```
C001                    ASMB,R,L,C
**  NO ERRORS PASS#1 **RTE ASMB 92067-16011**
```

```
C001                    ASMB,R,L,C
C002*
C003  00000                     NAM BASIC
C004*
C005**
C006*****************DUSM SUPER BASIC*********************
C007**
C008*
C009***
C010*****  SPECIAL SWITCH REGISTER FUNCTIONS
C011***
C012**
C013*       SW 15   1       ALL OUTPUT ON LINE PRINTER
C014*               0       ALL OUTPUT ON SYSTEM TTY
C015*       SW 14   1       TERMINATE EXECUTION OR LISTING
C016*       SW 15   1       DELETE ALL REMARKS
C017*
C018                         COM PGPTR,PGPT1,PGPT2,PGPT,BGNPG
C019                         COM PG1,PG2,PBUFF,PBPTR,FTRAK
C020                         COM BFBL
C021*
C022*
C023*     PGPTR              TEMP VIRTUAL ADDRESS POINTER
C024*     PGPT1              PAGE 1 VIRTUAL ADDRESS POINTER
C025*     PGPT2              PAGE 2 VIRTUAL ADDRESS POINTER
C026*     PGPT               ADDRESS OF CURRENT PAGE POINTER
C027*     BGNPG              TEMP POINTER TO START OF CURRENT CORE RES. PAGE
C028*     PG1                START OF CORE PAGE #1
C029*     PG2                START OF CORE PAGE #2
C030*     PBUFF              FIRST WORD OF USERS PROGAM
C031*     PBPTR              LAST WORD+1 OF USERS PROGRAM
C032*     FTRAK              START OF WORK AREA USED FOR IM CODE STORAGE
C033*      BFBL               CORE BUFFER LENGTH (MULTIPLE # OF SECTORS)
C034  00000 014607R LWBM   DEF LWAMA
C035  00001 000000         BSS 1
C036  00002 000002  FWAM   ABS 2        FIRST WORD PROGRAM MEMORY
C037  00003 077777  LWAM   ABS 32767     LAST WORD PROGRAM MEMORY
C038  00004 00001C PT1    DEF PGPT1     ADDR OF V.A. POINTER FOR PAGE 1
C039  00005 00002C PT2    DEF PGPT2     ADDR OF V.A. POINTER FOR PAGE 2
C040  00006 000000  FCORE  BSS 1          START OF FREE CORE
C041  00007 000000  FDISK  BSS 1         START OF FREE DISK STORAGE
C042  00010 000000  SYMTF  BSS 1          START OF SYMBOL TABLE
C043  00011 014751R SYMTA  DEF SYMTB     SYMBOL TABLE END
C044  00012 000000  LSTAK  BSS 1          LOW-CORE STACK ADDRESS
C045  00013 013513R ASBTB  DEF FINIS
C046  00014 013313R SBTBE  DEF FINIS
C047  00015 000000  OLDLN  BSS 1   PREVIOUS LINE #
C048  00016 014751R .BUFA  DEF SYMTB
C049  00017 000000  BADDR  BSS 1          I/O BUFFER
C050  00020 000000  CCNT   BSS 1            POINTERS
C051  00021 014610R SBUFA  DEF SBUFB
C052  00022 000000  SBPTR  BSS 1          SYNTAX BUFFER POINTER
C053  00023 000000  TFLAG  BSS 1
C054  00024 000000  TTYFL  BSS 1
C055  00025 000000  ENFLG  BSS 1
```

BFRL

```
C056  00026  000000  TSTPT  BSS  1          TEMPORARY STACK POINTER
C057  00027  000000  LSTPT  BSS  1          LOW-CORE STACK POINTER
C058  00030  000000  HSTPT  BSS  1          HIGH-CORE STACK POINTER
C059  00031  000000  PRADD  BSS  1          PROGRAM EXECUTION
C060  00032  000000  NXTST  BSS  1            SEQUENCING INFORMATION
C061  00033  000000  .LNUM  BSS  1          CURRENT LINE NUMBER
C062  00034  000000  TYPE   BSS  1          CURRENT STATEMENT TYPE
C063  00035  000000  DSTRT  BSS  1          DATA
C064  00036  000000  NXTDT  BSS  1            STATEMENT
C065  00037  000000  DCCNT  BSS  1              POINTERS
C066  00040  000000  RSYM   BSS  1
C067  00041  000000  SIGN   BSS  1
C068  00042  000000  EXP    BSS  1
C069  00043  000000  XH     BSS  1          RANDOM
C070  00044  000000  XL     BSS  1            VARIABLE
C071  00045  000000  TEMPS  BSS  12         TEMPORARIES
C072  00057         MLBX1  EQU  TEMPS+10
C073  00061  000000  B1     BSS  2
C074  00063  000000  B2     BSS  2
C075  00065  000000  B3     BSS  2
C076  00067  011077R .LOGA  DEF  .LOG
C077  00070  015274R EOF    JSB  ERROR
C078  00071  015274R NOEOF  JSB  ERROR
C079  00072  010007R EBUFA  DEF  EBUFF
C080  00073  010012R EBFA   DEF  EBFF-1
C081  00074  010015R LBUFA  DEF  LBUFF
C082  00075  010021R LNBFA  DEF  LNBFF-1
C083  00076  007673R ERBS   DEF  ERR-1
C084  00077  000100  RECER  DEF  RCERR-ERR
C085  00100  002225R FOPBS  DEF  QUOTE-2
C086  00101  103041R STBAS  DEF  SYNTB-26,I
C087  00102  106114R XECBR  DEF  XECTB-26,I
C088  00103  107111R ARBAS  DEF  ARUTB-6,I
C089  00104  010024R PDFBS  DEF  PDFT-1
C090  00105  003765R TBLAD  DEF  SYCMD
C091  00106  004025R STTYP  DEF  LET
C092  00107  004106R MATIO  DEF  READ
C093  00110  004144R MCBOP  DEF  AND
C094  00111  004162R PDFNS  DEF  SIN
C095  00112  004223R MATFN  DEF  ZER
C096  00113  004054R ANEXT  DEF  NEXT
C097  00114  004067R AEND   DEF  END
C098  00115  004103R ADATA  DEF  DATA
C099  00116  004131R ATHEN  DEF  THEN
C100  00117  004134R ATO    DEF  TO
C101  00120  004136R ASTEP  DEF  STEP
C102  00121  004141R ANOT   DEF  NOT
C103  00122  004157R ATAB   DEF  TAB
C104  00123  000057R MBXL   DEF  MLBX1
C105  00124  013313R BGNTB  DEF  FINIS      START OF CALL LINKAGE TABLE
C106  00125  013313R ENDTB  DEF  FINIS      LAST WORD +1 OF CALL TABLE
C107  00000         A      EQU  0
C108  00001         B      EQU  1
C109  00053         .0     EQU  53B
C110  00054         .1     EQU  .0+1
```

```
C111  00055          .2     EQU .0+2
C112  00056          .3     EQU .0+3
C113  00057          .4     EQU .0+4
C114  00060          .5     EQU .0+5
C115  00061          .6     EQU .0+6
C116  00062          .7     EQU .0+7
C117  00063          .8     EQU .0+8
C118  00064          .9     EQU .0+9
C119  00065          .10    EQU .0+10
C120  00126 000013   .11     DEC 11
C121  00127 000014   .12     DEC 12
C122  00130 000016   .14     DEC 14
C123  00070          .15    EQU .0+13
C124  00131 000022   .18    DEC 18
C125  00132 000025   .21    DEC 21
C126  00133 000026   .22    DEC 22
C127  00134 000027   .23    DEC 23
C128  00135 000032   .26    DEC 26
C129  00136 000033   .27    DEC 27
C130  00137 000034   .28    DEC 28
C131  00140 000036   .30    DEC 30
C132  00071          .31    EQU .0+14
C133  00141 000040   .32    DEC 32
C134  00142 000041   .33    DEC 33
C135  00143 000042   .34    DEC 34
C136  00144 000045   .37    DEC 37
C137  00145 000050   .40    DEC 40
C138  00146 000051   .41    DEC 41
C139  00147 000053   .43    DEC 43
C140  00150 000055   .45    DEC 45
C141  00151 000054   .44    DEC 44
C142  00152 000056   .46    DEC 46
C143  00153 000057   .47    DEC 47
C144  00154 000060   .48    DEC 48
C145  00155 000061   .49    DEC 49
C146  00156 000062   .50    DEC 50 **::
C147  00157 000072   .58    DEC 58
C148  00072          .63    EQU .0+15
C149  00160 000100   B100   OCT 100
C150  00161 000105   E      OCT 105
C151  00162 000106   F      OCT 106
C152  00163 000110   .72    DEC 72
C153  00164 000112   .74    DEC 74
C154  00165 000113   .75    DEC 75
C155  00166 000116   N      OCT 116
C156  00167 000123   S      OCT  123
C157  00072          B77    EQU .63
C158  00170 000133   B133   OCT 133
C159  00171 000137   B137   OCT 137
C160  00073          B177   EQU .0+16
C161  00172 000200   B200   OCT 200
C162  00173 000777   MSK1A  OCT 777 :::
C163  00074          MSK0   EQU .0+17
C164  00174 000400   B400   OCT 400
C165  00175 000776   B776   OCT 776
```

```
C166  00176  000377  MSK1   OCT 377 ::
C167  00177  001000  B1000  OCT 1000 ::
C168  00200  001400  B1400  OCT 1400 ::
C169  00201  003002  SCCNT  OCT 3002
C170  00202  002000  B2000  OCT 2000 ::
C171  00203  002400  B2400  OCT 2400 ::
C172  00204  006000  B6000  OCT 6000 ::
C173  00205  010400  UNMNC  OCT 10400 ::
C174  00206  011000  B1100  OCT 11000 ::
C175  00207  011400  B1140  OCT 11400 ::
C176  00210  020000  B20K   OCT 20000
C177  00211  016400  DEFOP  OCT 16400 ::
C178  00212  017000  REMOP  OCT 17000 ::
C179  00213  025000  RDOP   OCT 25000 ::
C180  00214  031000  WROP   OCT 31000
C181  00215  025400  PRNOP  OCT 25400
C182  00216  005000  LF     OCT 5000
C183  00217  063146  TENTH  OCT 63146
C184  00220  077400  OPMSK  OCT 77400  ::
C185  00221  077600  MSK4   OCT 77600
C186  00222  077777  INF    OCT 77777
C187  00223  100007  TYPFL  OCT 100007
C188  00224  100017  TABCN  OCT 100017
C189  00225  100377  OPDMK  OCT 100377 ::
C190  00226  130000  RMODE  OCT 130000
C191  00227  140000  UNNRM  OCT 140000
C192  00230  174000  HIMSK  OCT 174000
C193  00231  100000  N1     DEC -1.0
      00232  000000
C194  00052         M1     EQU .0-1
C195  00051         M2     EQU .0-2
C196  00050         M3     EQU .0-3
C197  00047         M4     EQU .0-4
C198  00046         M5     EQU .0-5
C199  00045         M6     EQU .0-6
C200  00044         M7     EQU .0-7
C201  00043         M8     EQU .0-8
C202  00042         M9     EQU .0-9
C203  00041         M10    EQU .0-10
C204  00233  177765  M11    DEC -11
C205  00234  177764  M12    DEC -12
C206  00235  177761  M15    DEC -15
C207  00236  177760  M16    DEC -16
C208  00237  177753  M21    DEC -21
C209  00240  177740  M32    DEC -32
C210  00241  177725  D53    OCT -53
C211  00242  177716  D62    OCT -62
C212  00243  177715  D65    OCT -65
C213  00244  177770  SBCON  OCT 177770
C214  00245  100001  STCON  OCT 100001
C215  00246  177706  D72    OCT -72
C216  00077         D100   EQU .0+20
C217  00247  177670  M72    DEC -72
C218  00250  177667  M73    DEC -73
C219  00251  177664  M76    DEC -76
```

```
C220  00252 177645  D133   OCT -153
C221  00253 177600  D200   OCT -200
C222  00075         M256   EQU .0+18
C223  00254 177312  M310   DEC -310
C224  00255 176030  M1000  DEC -1000
C225  00256 100000  MAXSN  DEC -32768
C226  00044         MSK3   EQU M7
C227  00257 043116  FN     ASC 1,FN
C228  00260 037440  QMARK  ASC 1,?
C229  00261 040000  HALF   OCT 40000
C230  00262 000000         OCT 0
C231  00261         HONE   EQU HALF
C232  00263 100000  MNEG   OCT 100000    MAXIMUM NEGATIVE FLOATING
C233  00264 000376         OCT 376            POINT NUMBER
C234  00263         FLGBT  EQU MNEG
C235  00265 102756  MAXFX  DEC -999999.5
      00266 002050
C236  00267 114631  MINFX  DEC -0.099999959
      00270 116573
C237  00271 000040  BLANK  OCT 40
C238  00272 000000  PARM   BSS 1
C239  00273 177755  TBLX   DEC -21
```

```
C241**
C242***  EMIT ERROR MESSAGE  ** '
C243**
C244   00274 000000  ERROR NOP
C245   00275 063741R        LDA PRNTL       SHIFT TO
C246   00276 073131R        STA LISTR       COMMAND MODE
C247   00277 062020R        LDA CCNT      SAVE
C248   00300 073266R        STA UPCHK       OUTPUT
C249   00301 062017R        LDA BADDR         BUFFER
C250   00302 073321R        STA RSCHK           POINTERS
C251   00303 062073R        LDA EBFA      SET BUFFER
C252   00304 072017R        STA BADDR       POINTER
C253   00305 060063         LDA .8        SET CHARACTER
C254   00306 072020R        STA CCNT        COUNT
C255   00307 066274R        LDB ERROR     ERROR SOURCE IN (B)
C256   00310 062076R        LDA ERBS      ERROR ADDRESS IN (A)
C257   00311 002004         INA           MOVE TO NEXT ERROR
C258   00312 154000         CPB 0,1       SAME AS ACTUAL ERROR?
C259   00313 003005         CMA,INA,RSS   YES
C260   00314 026311R        JMP *-3       NO
C261   00315 042076R        ADA ERBS      COMPUTE ERROR
C262   00316 073426R        STA ENOUT     SAVE NEGATIVE OF ERROR
C263   00317 003004         CMA,INA           NUMBER
C264   00320 017132R        JSB OUTIN    NUMBER TO BUFFER
C265   00321 066072R        LDB EBUFA     LOAD BUFFER ADDRESS
C266   00322 062020R        LDA CCNT      LOAD NEGATIVE OF
C267   00323 017161R        JSB EXECW    OUTPUT ERROR MESSAGE
C268   00324 062075R        LDA LNBFA     OUTPUT
C269   00325 072017R        STA BADDR
C270   00326 060065         LDA .10
C271   00327 072020R        STA CCNT        LINE
C272   00330 062033R        LDA .LNUM
C273   00331 017132R        JSB OUTIN
C274   00332 066074R        LDB LBUFA           NUMBER
C275   00333 062020R        LDA CCNT
C276   00334 017161R        JSB EXECW
C277   00335 063426R        LDA ENOUT     RETRIEVE NEGATIVE OF ERROR
C278   00336 042077R        ADA RECER     RECOVERABLE
C279   00337 002021         SSA,RSS
C280   00340 027611R        JMP PEXMK   NO, RETURN TO SYNTAX MODE
C281   00341 040061         ADA .6
C282   00342 002020         SSA
C283   00343 026340R        JMP *-3
C284   00344 062226R        LDA RMODE     RETURN TO
C285   00345 073131R        STA LISTR       RUN MODE
C286   00346 063266R        LDA UPCHK     RESTORE
C287   00347 072020R        STA CCNT        OUTPUT
C288   00350 063521R        LDA RSCHK         BUFFER
C289   00351 072017R        STA BADDR           POINTERS
C290   00352 126274R        JMP ERROR,1   RETURN TO PROGRAM
```

```
C292**
C293*** MOVE WORDS TO HIGHER DISK
C294**
C295  00353 000000  MVTOH NOP
C296  00354 066050R       LDB TEMP2      FETCH SOURCE ADDRESS
C297  00355 056051R MVT01 CPB TEMP3      ALL RELOCATION DONE?
C298  00356 126353R       JMP MVTOH,I    YES, EXIT
C299  00357 003400        CCA            BACK UP
C300  00360 042052R       ADA TEMP4        SOURCE AND
C301  00361 072052R       STA TEMP4          .DESTINATION
C302  00362 044052        ADB M1               ADDRESSES
C303  00363 017051R       JSB LDBI     MOVE WORD
C304  00364 017151R       JSB DSTOR            WORD
C305  00365 000052R       DEF TEMP4
C306  00366 026355R       JMP MVT01
C307**
C308*** MOVE WORDS TO HIGHER CORE
C309**
C310  00367 000000  MVTOC NOP
C311  00370 066050R       LDB TEMP2
C312  00371 056051R MVT1  CPB TEMP3
C313  00372 126367R       JMP MVTOC,I
C314  00373 003400        CCA
C315  00374 042052R       ADA TEMP4
C316  00375 072052R       STA TEMP4
C317  00376 044052        ADB M1
C318  00377 160001        LDA 1,I
C319  00400 172052R       STA TEMP4,I
C320  00401 026371R       JMP MVT1
C321**
C322***  INPUT A CONSTANT  **
C323**
C324  00402 000000  CONST NOP
C325  00403 017377R       JSB GETCR
C326  00404 126402R       JMP CONST,I
C327  00405 006400        CLB            SET SIGN
C328  00406 076041R       STB SIGN         POSITIVE
C329  00407 005004        INB
C330  00410 052147R       CPA .43        '+' ?
C331  00411 026415R       JMP CONS1      YES
C332  00412 052150R       CPA .45        NO, '-' ?
C333  00413 007401        CCB,RSS        YES
C334  00414 026420R       JMP CONS2      NO
C335  00415 076041R CONS1 STB SIGN       RECORD SIGN
C336  00416 017377R       JSB GETCR      FETCH NEXT
C337  00417 026426R       JMP SYE12-1      CHARACTER
C338  00420 015430R CONS2 JSB NUMCK       FETCH CONSTANT
C339  00421 026424R       JMP CONS3      NONE FOUND
C340  00422 036402R       ISZ CONST      SUCCESSFULLY FOUND,
C341  00423 126402R       JMP CONST,I      EXIT VIA (P+2)
C342  00424 056041R CONS3 CPB SIGN       SIGN FOUND? ( (B) = 0)
C343  00425 003401        CCA,RSS        NO
C344  00426 015274R       JSB ERROR      YES, SOLITARY SIGN
C345  00427 126402R SYE12 JMP CONST,I    EXIT VIA (P+1)
C346**
```

```
C347***  FETCH NUMBER AND CONVERT TO BINARY  **
C348**
C349  00430 000000  NUMCK NOP            CHARACTER IN (A), SIGN SET
C350  00431 005400        CLB
C351  00432 076042R       STB EXP        ZERO
C352  00433 077066R       STB MANT1        ALL
C353  00434 077130R       STB MANT2          COMPONENTS
C354  00435 077366R       STB EXPON            OF NUMBER
C355  00436 076051R       STB TEMP3      SET 'NUMBER' FLAG FALSE
C356  00437 007400        CCB            SET 'DECIMAL POINT'
C357  00440 077416R       STB DPFLG        FLAG FALSE
C358  00441 052152R NUMC1 CPA .46        DECIMAL POINT?
C359  00442 037416R       ISZ DPFLG      YES, SET FLAG TRUE
C360  00443 026447R       JMP NUMC2      NO
C361  00444 002400        CLA            INITIALIZE POST-DECIMAL DIGIT
C362  00445 073366R       STA EXPON        DIGIT COUNTER TO ZERO
C363  00446 026466R       JMP NUMC3+1    FETCH A CHARACTER
C364  00447 017353R NUMC2 JSB DIGCK      DIGIT?
C365  00450 026521R       JMP NUMC7      NO
C366  00451 037366R       ISZ EXPON      YES, COUNT DIGIT
C367  00452 001727        ALF,ALF        LEFT-JUSTIFY
C368  00453 001723        ALF,RAR          DIGIT AND
C369  00454 072052R       STA TEMP4          SAVE IT
C370  00455 016777R       JSB MBY10      MULTIPLY PREVIOUS NUMBER BY 10
C371  00456 066042R       LDB EXP
C372  00457 006002        SZB            ZERO EXPONENT?
C373  00460 026472R       JMP NUMC4      NO
C374  00461 060057        LDA .4         YES, SET
C375  00462 072042R       STA EXP          EXPONENT TO 4
C376  00463 062052R       LDA TEMP4      LOAD
C377  00464 005400        CLB              NUMBER
C378  00465 016742R NUMC3 JSB NORML      NORMALIZE THE NUMBER
C379  00466 036051R       ISZ TEMP3      SET 'NUMBER OCCURRED' FLAG
C380  00467 017377R       JSB GETCR      ANOTHER CHARACTER?
C381  00470 026570R       JMP NUM12      NO
C382  00471 026441R       JMP NUMC1      YES
C383  00472 044047  NUMC4 ADB M4         COMPUTE
C384  00473 007000        CMB              EXPONENT
C385  00474 062052R       LDA TEMP4          BIAS AND
C386  00475 076052R       STB TEMP4            SAVE IT
C387  00476 005400        CLB
C388  00477 036052R NUMC5 ISZ TEMP4      DIGIT POSITIONED?
C389  00500 026516R       JMP NUMC6      NO
C390  00501 000040        CLE            YES, ADD IN
C391  00502 047130R       ADB MANT2        LOW PART
C392  00503 103101        CLO                OF NUMBER
C393  00504 002040        SEZ            OVERFLOW?
C394  00505 002004        INA            YES, BUMP (A)
C395  00506 043066R       ADA MANT1      ADD IN HIGH PART OF NUMBER
C396  00507 102501        SOS            OVERFLOW?
C397  00510 026465R       JMP NUMC3      NO
C398  00511 000065        CLE,ERA        YES, ROTATE
C399  00512 005500        ERB              DOWN AND
C400  00513 036042R       ISZ EXP            BUMP
C401  00514 000000        NOP                EXPONENT
```

```
C402  00515 026465R         JMP NUMC3
C403  00516 000065  NUMC6  CLE,ERA        SHIFT
C404  00517 005500         ERB              DIGIT
C405  00520 026477R         JMP NUMC5        RIGHT
C406  00521 006400  NUMC7  CLB             DECIMAL POINT
C407  00522 076052R         STB TEMP4
C408  00523 056051R         CPB TEMP3        OR DIGIT FOUND?
C409  00524 126430R         JMP NUMCK,I      NO, EXIT VIA (P+1)
C410  00525 052161R         CPA E            YES, 'E' ?
C411  00526 002001          RSS              YES
C412  00527 026570R         JMP NUM12        NO, NO EXPONENT PART
C413  00530 017377R         JSB GETCR
C414  00531 016274R NUMER  JSB ERROR
C415  00532 052147R         CPA .43          '+' ?
C416  00533 026540R         JMP NUMC8        YES
C417  00534 052150R         CPA .45          NO, '-' ?
C418  00535 003401          CCA,RSS          YES
C419  00536 026542R         JMP NUMC9        NO
C420  00537 072052R         STA TEMP4        NOTE MINUS SIGN
C421  00540 017377R NUMC8  JSB GETCR
C422  00541 026531R         JMP NUMER
C423  00542 017353R NUMC9  JSB DIGCK        DIGIT?
C424  00543 026531R         JMP NUMER        NO
C425  00544 072051R         STA TEMP3        YES, SAVE IT
C426  00545 017377R         JSB GETCR
C427  00546 026564R         JMP NUM10        SECOND
C428  00547 017353R         JSB DIGCK          DIGIT?
C429  00550 026564R         JMP NUM10        NO
C430  00551 066051R         LDB TEMP3        YES
C431  00552 005020          BLS,BLS          MULTIPLY
C432  00553 046051R         ADB TEMP3          PRIOR DIGIT
C433  00554 005000          BLS                BY 10
C434  00555 040001          ADA 1            ADD NEW DIGIT
C435  00556 072051R         STA TEMP3        SAVE EXPONENT
C436  00557 017377R         JSB GETCR
C437  00560 026564R         JMP NUM10        THIRD
C438  00561 017353R         JSB DIGCK          DIGIT?
C439  00562 002001          RSS              NO
C440  00563 026531R         JMP NUMER        YES
C441  00564 062051R NUM10  LDA TEMP3        LOAD EXPONENT
C442  00565 056052R         ISZ TEMP4        POSITIVE?
C443  00566 003004          CMA,INA          YES, COMPLEMENT IT
C444  00567 002001          RSS              NO
C445  00570 002400  NUM12  CLA              CLEAR IF NO EXPONENT PART
C446  00571 037416R         ISZ DPFLG        DECIMAL POINT?
C447  00572 043366R         ADA EXPON        YES, CORRECT EXPONENT
C448  00573 002003          SZA,RSS          ZERO EXPONENT?
C449  00574 026611R         JMP NUM14        YES
C450  00575 002020          SSA              NO, NEGATIVE EXPONENT?
C451  00576 026605R         JMP NUM13        NO
C452  00577 003004          CMA,INA          YES, SET
C453  00600 073366R         STA EXPON          COUNTER
C454  00601 017030R         JSB DBY10        DIVIDE NUMBER BY 10
C455  00602 037366R         ISZ EXPON        DONE?
C456  00603 026601R         JMP *-2          NO
```

```
C457  00604 026611R        JMP NUM14      YES
C458  00605 073566R NUM13  STA EXPON      SET COUNTER
C459  00606 015777R        JSB MBY10      MULTIPLY BY 10
C460  00607 037566R        ISZ EXPON      DONE?
C461  00610 026606R        JMP *-2        NO
C462  00611 063066R NUM14  LDA MANT1      YES, LOAD
C463  00612 067130R        LDB MANT2        NUMBER
C464  00613 036041R        ISZ SIGN       POSITIVE?
C465  00614 026620R        JMP NUM15      YES
C466  00615 003000          CMA            NO,
C467  00616 007007          CMB,INB,SZB,RSS   COMPLEMENT
C468  00617 002004          INA                IT
C469  00620 015647R NUM15  JSB .PACK      PACK NUMBER INTO (A) AND (B)
C470  00621 104400          DST SVOP1
      00622 015516R
C471  00623 062022R        LDA SBPTR
C472  00624 002020          SSA
C473  00625 026642R        JMP NUM17
C474  00626 104200          DLD SVOP1
      00627 015316R
C475  00630 036022R        ISZ SBPTR
C476  00631 172022R        STA SBPTR,I    STORE
C477  00632 036022R        ISZ SBPTR        NUMBER IN
C478  00633 176022R        STB SBPTR,I        PROPER
C479  00634 036022R        ISZ SBPTR            LOCATION
C480  00635 017416R NUM16  JSB BCKSP      FETCH
C481  00636 017577R        JSB GETCR        FIRST
C482  00637 060065          LDA .10            UNUSED CHARACTER
C483  00640 036430R        ISZ NUMCK      RETURN
C484  00641 126430R        JMP NUMCK,I      VIA (P+2)
C485  00642 104200  NUM17  DLD SVOP1
      00643 015316R
C486  00644 017342R        JSB DDST
C487  00645 000022R        DEF SBPTR
C488  00646 026635R        JMP NUM16
C489**
C490***  NORMALIZE AND PACK FLOATING POINT NUMBER  **
C491**
C492  00647 000000  .PACK  NOP            MANTISSA IN (A) AND (B),
C493  00650 015742R        JSB NORML        EXPONENT IN EXP, (E) CLEARED
C494  00651 002103          CLE,SZA,RSS    ZERO RESULT?
C495  00652 126647R        JMP .PACK,I    YES
C496  00653 044073          ADB B177       NO, ROUND
C497  00654 002021          SSA,RSS        POSITIVE NUMBER?
C498  00655 006004          INB            YES, FINISH ROUND
C499  00656 103101          CLO
C500  00657 002040          SEZ            OVERFLOW FROM (B)?
C501  00660 002104          CLE,INA        YES, BUMP (A)
C502  00661 102301          SOS            OVERFLOW? (A=100000, B=0)
C503  00662 001200          RAL
C504  00663 002031          SSA,SLA,RSS    TWO HIGH BITS 1'S? (A=140000)
C505  00664 026667R        JMP PACK1      NO
C506  00665 002300          CCE            YES
C507  00666 001130          ARS,SLA,ALS    SET (A) =100000 AND SKIP
C508  00667 001300  PACK1  RAR            COUNTERPART TO *-5
```

```
C509  00670 072777R        STA MBY10     SAVE (A)
C510  00671 060001         LDA 1         DELETE 8 LOW
C511  00672 010075         AND M256        ORDER BITS OF MANTISSA
C512  00673 070001         STA 1         SAVE LOWER MANTISSA
C513  00674 062042R        LDA EXP       FETCH EXPONENT
C514  00675 002040         SEZ           DECREMENT EXPONENT?
C515  00676 040052         ADA M1        YES
C516  00677 102201         SOC           NO, PRIOR OVERFLOW?
C517  00700 002004         INA           YES, INCREMENT EXPONENT
C518  00701 042172R        ADA B200      NO, EXPONENT
C519  00702 002020         SSA             UNDERFLOW?
C520  00703 026722R        JMP PACK3     YES
C521  00704 040075         ADA M256      NO, EXPONENT
C522  00705 002021         SSA,RSS         OVERFLOW?
C523  00706 026726R        JMP PACK4     YES
C524  00707 042172R        ADA B200      NO, RESTORE EXPONENT,
C525  00710 001200         RAL             POSITION SIGN,
C526  00711 010074         AND MSK0          MASK TO 8 BITS, AND
C527  00712 044000         ADB 0             COMBINE WITH LOW MANTISSA
C528  00713 062777R        LDA MBY10     RETRIEVE HIGH MANTISSA
C529  00714 052263R        CPA MNEG
C530  00715 002001         RSS           NEGATIVE
C531  00716 125647R        JMP .PACK,I
C532  00717 056264R        CPB MNEG+1      OVERFLOW?
C533  00720 026726R        JMP PACK4     YES
C534  00721 125647R        JMP .PACK,I   NO
C535  00722 015274R PACK3  JSB ERROR
C536  00723 002400  UNDER  CLA           ZERO RESULT
C537  00724 006400         CLB             ON UNDERFLOW
C538  00725 125647R        JMP .PACK,I
C539  00726 015274R PACK4  JSB ERROR
C540  00727 062777R OVRER  LDA MBY10
C541  00730 015732R        JSB OVFLW
C542  00731 125647R        JMP .PACK,I
C543**
C544***  LOAD INFINITY ON OVERFLOW  **
C545**
C546  00732 000000  OVFLW  NOP
C547  00733 064051         LDB M2        LOAD
C548  00734 002020         SSA             APPROPRIATE
C549  00735 066175R        LDB B776          LOW MANTISSA
C550  00736 032222R        IOR INF       LOAD
C551  00737 002020         SSA             APPROPRIATE
C552  00740 062263R        LDA MNEG          HIGH MANTISSA
C553  00741 125732R        JMP OVFLW,I
C554**
C555***  NORMALIZE (A), (B), AND EXP  **
C556**
C557  00742 000000  NORML  NOP           SET
C558  00743 072777R        STA MBY10       LEFT-SHIFT
C559  00744 002400         CLA               COUNTER
C560  00745 072776R        STA MPY             TO ZERO
C561  00746 062777R        LDA MBY10
C562  00747 002003         SZA,RSS       ON
C563  00750 006002         SZB             ZERO
```

```
C564  00751 026757R        JMP NORM3          CLEAR
C565  00752 072042R        STA EXP              EVERYTHING
C566  00753 073066R        STA MANT1        STORE
C567  00754 077130R NORM1  STB MANT2          MANTISSA
C568  00755 126742R        JMP NORML,I          AND RETURN
C569  00756 036776R NORM2  ISZ MPY          COUNT LEFT SHIFTS
C570  00757 004066  NORM3  CLE,ELB          ROTATE (A) AND
C571  00760 001600        ELA                  (B) LEFT INTO (E)
C572  00761 002061        SEZ,SSA,RSS      TWO HIGHEST BITS 0?
C573  00762 026756R        JMP NORM2        YES, + UNNORMALIZED
C574  00763 002060        SEZ,SSA          NO, TWO HIGHEST BITS 1?
C575  00764 026756R        JMP NORM2        YES, - UNNORMALIZED
C576  00765 001500        ERA              SHIFT TO
C577  00766 005540        ERB,CLE            NORMALIZE MANTISSA
C578  00767 073066R        STA MANT1        NO,
C579  00770 062776R        LDA MPY            COMPUTE
C580  00771 003004        CMA,INA              CORRECTED
C581  00772 042042R        ADA EXP                EXPONENT
C582  00773 072042R        STA EXP                  VALUE
C583  00774 063066R        LDA MANT1
C584  00775 026754R        JMP NORM1
C585  00776 000000  MPY    BSS 1
C586**
C587***  MULTIPLY UNPACKED NUMBER BY 10  **
C588**
C589  00777 000000  MBY10  NOP
C590  01000 063066R        LDA MANT1        RETURN ON
C591  01001 002003        SZA,RSS            ZERO
C592  01002 125777R        JMP MBY10,I          MANTISSA
C593  01003 066042R        LDB EXP          MULTIPLY
C594  01004 044056        ADB .3             BY
C595  01005 076042R        STB EXP              8
C596  01006 067130R        LDB MANT2        LOAD MANTISSA
C597  01007 000065        CLE,ERA          DIVIDE
C598  01010 005500        ERB                BY
C599  01011 000065        CLE,ERA              4
C600  01012 005540        ERB,CLE
C601  01013 047130R        ADB MANT2        DOUBLE
C602  01014 002040        SEZ                ADD TO
C603  01015 002004        INA                  PRODUCE
C604  01016 043066R        ADA MANT1              1.25 * MANTISSA
C605  01017 002021        SSA,RSS          CORRECT
C606  01020 027025R        JMP *+5
C607  01021 000065        CLE,ERA            ON
C608  01022 005500        ERB
C609  01023 036042R        ISZ EXP              OVERFLOW
C610  01024 000000        NOP
C611  01025 073066R        STA MANT1
C612  01026 077130R        STB MANT2
C613  01027 125777R        JMP MBY10,I
C614**
C615***  DIVIDE UNPACKED NUMBER BY 10  **
C616**
C617  01030 000000  DBY10  NOP              MULTIPLY BY DOUBLE-LENGTH TENTH
C618  01031 063066R        LDA MANT1        RETURN
```

```
C619  01032 002003        SZA,RSS          ON ZERO
C620  01033 127030R       JMP DBY10,I        MANTISSA
C621  01034 064051        LDB M2         ADD EXPONENT OF
C622  01035 046042R       ADB EXP          'TENTH' TO
C623  01036 075042R       STB EXP            MANTISSA EXPONENT
C624  01037 065150R       LDA MANT2      JUSTIFY
C625  01040 000065        CLE,ERA          LOWER MANTISSA
C626  01041 100200        MPY TENTH      MULTIPLY BY 63146 (ONE-TENTH)
      01042 000217R
C627  01043 000006        CLE,ELA        SHIFT
C628  01044 005640        ELB,CLE          BACK
C629  01045 040001        ADA 1          ADD IN LOWER MANTISSA*
C630  01046 002040        SEZ              TENTH*(2)-16
C631  01047 006004        INB                AND ROUND
C632  01050 077150R       STB MANT2            TO 16 BITS
C633  01051 063066R       LDA MANT1      DO
C634  01052 100200        MPY TENTH      SAVE FOR
      01053 000217R
C635  01054 000040        CLE                  HIGH
C636  01055 040001        ADA 1                  MANTISSA
C637  01056 043150R       ADA MANT2      (EFFECTIVELY) SUM
C638  01057 002040        SEZ              DOUBLE-LENGTH
C639  01060 006004        INB                PRODUCTS
C640  01061 077066R       STB MANT1      EXCHANGE
C641  01062 070001        STA 1            (A) AND (B)
C642  01063 063066R       LDA MANT1          REGISTERS
C643  01064 016742R       JSB NORML      NORMALIZE RESULT
C644  01065 127030R       JMP DBY10,I
```

```
C646**
C647***  FIND AND STORE ONE-CHARACTER OPERATORS  **
C648**
C649  01066 000000  SYMCK NOP             CHARACTER IN (A)
C650  01067 075053R       STB COUNT       -(ENTRIES TO BE SEARCHED)
C651  01070 001727       ALF,ALF         POSITION
C652  01071 032141R       IOR .32           CHARACTER
C653  01072 167066R       LDB SYMCK,I     STARTING TABLE ENTRY - 2
C654  01073 037066R       ISZ SYMCK       SET RETURN ADDRESS
C655  01074 044055  SYMC1 ADB .2          UPDATE TABLE POINTER
C656  01075 150001       CPA 1,I         MATCH?
C657  01076 027104R       JMP SYMC2
C658  01077 035053R       ISZ COUNT       NO, CONTINUE SEARCH?
C659  01100 027074R       JMP SYMC1       YES
C660  01101 001727       ALF,ALF         NO, RESTORE
C661  01102 010073       AND B177          CHARACTER
C662  01103 127066R       JMP SYMCK,I         AND EXIT
C663  01104 005400  SYMC2 CCA             GET
C664  01105 040001       ADA 1             INFORMATION
C665  01106 160000       LDA 0,I             WORD
C666  01107 012220R       AND UPMSK             AND
C667  01110 172022R       STA SBPTR,I             STORE IT
C668  01111 052204R       CPA B6000  ::
C669  01112 027323R       JMP FSC14
C670  01113 037066R       ISZ SYMCK       RETURN VIA
C671  01114 127066R       JMP SYMCK,I       (P+2)
C672**
C673***  FIND CALLED SUBROUTINE  **
C674**
C675  01115 000000  FNDSB NOP
C676  01116 075050R       STB TEMP2       SAVE SUBROUTINE NUMBER
C677  01117 066124R       LDB BGNTB       LOAD (B) WITH SUBROUTINE TABLE
C678  01120 056125R FNDS1 CPB ENDTB       END OF TABLE?
C679  01121 016274R       JSB ERROR       YES
C680  01122 160001  CALER LDA 1,1         NO, EXTRACT
C681  01123 010072       AND .63           SUBROUTINE NUMBER
C682  01124 052050R       CPA TEMP2       DESIRED ONE?
C683  01125 127115R       JMP FNDSB,I     YES
C684  01126 044055       ADB .2          NO, MOVE TO
C685  01127 027120R       JMP FNDS1         NEXT TABLE ENTRY



C687*              **********************************************
C688*              SUBROUTINE TO COMPUTE THE STORAGE REQUIRED BY AN
C689*              ARRAY WHOSE PACKED DIMENSIONS ARE IN A UPON ENTRY
C690*              **********************************************
C691*
C692*              THE SUBROUTINE RETURNS IN A THE NUMBER OF LOCATIONS
C693*              REQUIRED FOR THE SPECIFIED DIMENSIONS
C694*              = 2*DIM1*DIM2
C695*
C696  01130 000000  MDIM  NOP
```

```
C697  01131 070001          STA 1          STORE PACKED DIMS. TEMPORALILY
C698  01132 010074          AND MSK0
C699  01133 073223R         STA .FLUN      STORE # OF COLUMNS
C700  01134 060001          LDA 1
C701  01135 001727          ALF,ALF
C702  01136 010074          AND MSK0       A = # OF ROWS
C703  01137 001000          ALS            DOUBLE FOR FLOATING POINT
C704  01140 100200          MPY .FLUN      COMPUTE 2*ROW*COLUMN
      01141 001223R
C705  01142 002020          SSA            RESULT < 32768 ?
C706  01143 016274R         JSB ERROR      NO, ERROR DIMENSIONS TOO LARGE
C707  01144 005002  MER9    SZB          RESULT < 64K?
C708  01145 027143R         JMP *-2      NO, ERROR
C709  01146 127130R         JMP MDIM,I   YES, RETURN
```

```
C711**
C712***  ROUND A SUBSCRIPT TO AN INTEGER  **
C713**
C714*
C715*  RETURNS INTEGER IN [1,32767] (BIASED BY -1)
C716*  OR EXITS TO ERROR.
C717*
C718  01147 000000  SBFIX NOP             SUBSCRIPT IN (A) AND (B)
C719  01150 017160R       JSB IFIX        24-BIT INTEGER?
C720  01151 027221R       JMP E6-1      NO
C721  01152 002041       SEZ,RSS         YES, ROUND AND
C722  01153 044052       ADB M1            BIAS BY -1
C723  01154 002003       SZA,RSS         15-BIT
C724  01155 006020       SSB               POSITIVE INTEGER?
C725  01156 027221R       JMP E6-1        NO
C726  01157 127147R       JMP SBFIX,I     YES
C727**
C728***  INTEGERIZE FLOATING POINT NUMBER  **
C729**
C730  01160 000000  IFIX  NOP
C731  01161 102101       STO
C732  01162 073377R       STA GETCR       SAVE (A)
C733  01163 017223R       JSB .FLUN       EXPONENT
C734  01164 002020       SSA               NON-NEGATIVE?
C735  01165 027210R       JMP IFIX3       NO
C736  01166 042236R       ADA M16         YES, EXPONENT
C737  01167 002020       SSA               <= 15?
C738  01170 103101       CLO             YES
C739  01171 040043       ADA M8          EXPONENT
C740  01172 002021       SSA,RSS           <= 23?
C741  01173 127160R       JMP IFIX,1      NO, ALL SIGNIFICANCE IS INTEGER
C742  01174 040043       ADA M8          MOVE BINARY POINT TO END OF (B)
C743  01175 073223R       STA .FLUN       SAVE SHIFT COUNT
C744  01176 063377R       LDA GETCR       RETRIEVE (A)
C745  01177 027204R       JMP IFIX2
C746  01200 000071  IFIX1 CLE,SLA,ARS     SHIFT (A) RIGHT
C747  01201 002200       CME             SHIFT (B)
C748  01202 004035       SLB,ERR           RIGHT
C749  01203 102101       STO             NOTE IF A 1 IS LOST
C750  01204 037223R IFIX2 ISZ .FLUN       DONE?
C751  01205 027200R       JMP IFIX1       NO
C752  01206 037160R       ISZ IFIX        YES
C753  01207 127160R       JMP IFIX,I
C754  01210 063377R IFIX3 LDA GETCR       RETRIEVE (A)
C755  01211 002120       CLE,SSA         TRUNCATE
C756  01212 003401       CCA,RSS           TO
C757  01213 002401       CLA,RSS             -1
C758  01214 007401       CCB,RSS               OR
C759  01215 006400       CLB                     0
C760  01216 027206R       JMP IFIX2+2
```

```
C762**
C763***  TAKE ARITHMETIC INVERSE  **
C764**
C765  01217 000000  ARINV NOP            NUMBER IN (A) AND (B)
C766  01220 016001X       FMP N1          MULTIPLY BY MINUS 1 TO TAKE INVERSE
      01221 000231R
C767  01222 127217R       JMP ARINV,I
C768**
C769***  UNPACK LOW WORD OF NUMBER  **
C770**
C771  01223 000000  .FLUN NOP            WORD IN (B)
C001  01224 060001       LDA 1           (A) = (B)
C002  01225 010074       AND MSK0        EXTRACT EXPONENT IN (A)
C003  01226 007000       CMB             SUBTRACT OFF
C004  01227 044000       ADB 0             EXPONENT FROM
C005  01230 007000       CMB                 MANTISSA IN (B)
C006  01231 000033       SLA,RAR         NEGATIVE EXPONENT?
C007  01232 032221R      IOR MSK4        YES, FILL IN LEADING BITS
C008  01233 127223R       JMP .FLUN,I    NO
C009**
C010***  STACK (B) ON LOW-CORE STACK  **
C011**
C012  01234 000000  SLWST NOP
C013  01235 036027R       ISZ LSTPT      ADVANCE 'LOW
C014  01236 062027R       LDA LSTPT        STACK' POINTER
C015  01237 052030R       CPA HSTPT      STACK OVERFLOW?
C016  01240 016274R E1    JSB ERROR      YES
C017  01241 176027R       STB LSTPT,I    NO, STACK (B)
C018  01242 127234R       JMP SLWST,I
```

```
C020**
C021***  BUMP HIGH STACK POINTER  **
C022**
C023  01243 000000  BHSTP NOP
C024  01244 007400        CCB             ADVANCE
C025  01245 046030R       ADB HSTPT
C026  01246 076030R       STB HSTPT         POINTER
C027  01247 056027R       CPB LSTPT       OVERFLOW?
C028  01250 027240R       JMP E1          YES
C029  01251 127243R       JMP BHSTP,I     NO
C030**
C031***  FETCH TOP OF STACK  **
C032**
C033  01252 000000  STTOP NOP
C034  01253 017266R       JSB OPCHK       VALIDATE
C035  01254 017321R       JSB RSCHK         OPERAND
C036  01255 166030R       LDB HSTPT,I     SAVE
C037  01256 006020         SSB         DATA ON DISK?
C038  01257 027263R       JMP STT01     YES
C039  01260 104200         DLD 1,1         LOAD NUMBER
      01261 100001
C040  01262 127252R       JMP STTOP,I
C041  01263 060001  STT01 LDA B
C042  01264 017505R       JSB FTHOP     LOAD NUMBER
C043  01265 127252R       JMP STTOP,I
C044**
C045***  VERIFY LEGITIMACY OF OPERAND  **
C046**
C047  01266 000000  OPCHK NOP
C048  01267 166030R       LDB HSTPT,I     OPERAND ADDRESS TO (B)
C049  01270 006020         SSB           DATA ON DISK?
C050  01271 017313R       JSB GETOP     YES, GET HIGH PART OF OPERAND
C051  01272 160001         LDA 1,1         HIGH PART OF
C052  01273 052263R       CPA MNEG          OPERAND 100000B?
C053  01274 006005         INB,RSS         YES
C054  01275 027304R       JMP OPCH1       NO
C055  01276 006020         SSB           DATA ON DISK?
C056  01277 017313R       JSB GETOP     YES, GET LOW PART OF OPERAND
C057  01300 160001         LDA 1,I         LOW PART
C058  01301 052264R       CPA MNEG+1        776B?
C059  01302 016274R .E8    JSB ERROR     YES
C060  01303 044052  E8     ADB M1
C061  01304 056026R OPCH1 CPB TSTPT       TEMPORARY OPERAND?
C062  01305 002001         RSS             YES
C063  01306 127266R       JMP OPCHK,I     NO
C064  01307 062026R       LDA TSTPT       UNSTACK
C065  01310 040051         ADA M2            THE TEMPORARY
C066  01311 072026R       STA TSTPT           OPERAND
C067  01312 127266R       JMP OPCHK,I     EXIT WITH ADDRESS IN (B)
C068  01313 000000  GETOP NOP
C069  01314 060001         LDA B
C070  01315 012222R       AND INF
C071  01316 017204R       JSB DREAD
C072  01317 037313R       ISZ GETOP
C073  01320 127313R       JMP GETOP,I
```

```
C074**
C075***  ALLOT SPACE FOR INTERMEDIATE RESULT  **
C076**
C077  01321 000000  RSCHK NOP
C078  01322 062026R       LDA TSTPT     ALLOT
C079  01323 040055       ADA .2
C080  01324 072026R       STA TSTPT       SPACE
C081  01325 040052       ADA M1        OVERFLOW INTO
C082  01326 052012R       CPA LSTAK       LOW-CORE STACK?
C083  01327 002001       RSS           YES
C084  01330 127321R       JMP RSCHK,I   NO
C085  01331 062012R       LDA LSTAK     SAVE
C086  01332 002004       INA             LOWER
C087  01333 072051R       STA TEMP3         STACK BOUND
C088  01334 040064       ADA .9        UPDATE
C089  01335 072012R       STA LSTAK       STACK BOTTOM
C090  01336 062027R       LDA LSTPT     SET
C091  01337 002004       INA             SOURCE
C092  01340 072050R       STA TEMP2         ADDRESS
C093  01341 040064       ADA .9        UPDATE
C094  01342 072027R       STA LSTPT       STACK TOP
C095  01343 002004       INA           SET .DESTINATION
C096  01344 072052R       STA TEMP4       ADDRESS
C097  01345 003004       CMA,INA       OVERFLOW
C098  01346 042030R       ADA HSTPT       INTO
C099  01347 002020       SSA               HIGH-CORE STACK?
C100  01350 027240R       JMP E1        YES
C101  01351 015367R       JSB MVTOC     NO, MOVE
C102  01352 127321R       JMP RSCHK,I     LOW-CORE STACK
C103**
C104***  CHECK FOR DIGIT  **
C105**
C106  01353 000000  DIGCK NOP           CHARACTER IN (A)
C107  01354 064000       LDB 0
C108  01355 046246R       ADB D72       ASCII 72B
C109  01356 006021       SSB,RSS         OR GREATER?
C110  01357 127353R       JMP DIGCK,I   YES, RETURN WITH CHARACTER
C111  01360 044065       ADB .10        NO, ASCII 60B
C112  01361 006020       SSB             OR GREATER?
C113  01362 127353R       JMP DIGCK,I   NO
C114  01363 037353R       ISZ DIGCK     YES, SET 'SUCCESS' EXIT,
C115  01364 060001       LDA 1           LOAD DIGIT INTO (A),
C116  01365 127353R       JMP DIGCK,I       AND RETURN
C117**
C118***  CHECK FOR LETTER  **
C119**
C120  01366 000000  LETCK NOP           CHARACTER IN (A)
C121  01367 064000       LDB 0
C122  01370 046252R       ADB D133      ASCII 133B
C123  01371 006021       SSB,RSS         OR GREATER?
C124  01372 127366R       JMP LETCK,I   YES, EXIT WITH CHARACTER IN (A)
C125  01373 046135R       ADB .26        NO, ASCII 101B
C126  01374 006021       SSB,RSS         OR GREATER?
C127  01375 037366R       ISZ LETCK      YES
C128  01376 127366R       JMP LETCK,I   NO
```

```
C129*
C130*  ON END-OF-FILE CONDITION RETURN TO P+1 ELSE
C131*  RETURN TO P+2 WITH NON-BLANK CHARACTER IN (A)
C132*
C133  01377 000000  GETCR NOP
C134  01400 055020R       ISZ CCNT       ANY CHARACTERS LEFT?
C135  01401 002001       RSS
C136  01402 127377R       JMP GETCR,I    NO, END-OF-FILE EXIT
C137  01403 066017R       LDB BADDR      LOAD BUFFER ADDRESS
C138  01404 055017R       ISZ BADDR      UPDATE FOR NEXT TIME
C139  01405 004065       CLE,ERB        SET CHARACTER FLAG
C140  01406 160001       LDA 1,I        LOAD CURRENT BUFFER WORD
C141  01407 002041       SEZ,RSS        FIRST CHARACTER?
C142  01410 001727       ALF,ALF        YES, POSITION IT
C143  01411 010073       AND B177       MASK EXTRANEOUS BITS
C144  01412 052271R       CPA BLANK      BLANK?
C145  01413 027400R       JMP GETCR+1    YES, FETCH NEXT CHARACTER
C146  01414 037377R       ISZ GETCR      UPDATE RETURN ADDRESS
C147  01415 127377R       JMP GETCR,I      AND EXIT
C148**
C149***  BACKSPACE OVER ONE CHARACTER  **
C150**
C151  01416 000000  BCKSP NOP
C152  01417 003400       CCA            BACKSPACE
C153  01420 042020R       ADA CCNT         OVER
C154  01421 072020R       STA CCNT           LAST
C155  01422 003400       CCA                  CHARACTER IN
C156  01423 042017R       ADA BADDR              INPUT
C157  01424 072017R       STA BADDR                BUFFER
C158  01425 127416R       JMP BCKSP,I
C159**
C160***  PRINT A NUMBER  **
C161**
C162  01426 000000  ENOUT NOP
C163  01427 002300       CCE            SET SIGN FLAG TRUE
C164  01430 015041R       JSB NUMOT      OUTPUT THE NUMBER
C165  01431 062141R       LDA .32        OUTPUT
C166  01432 017477R       JSB OUTCR        A BLANK
C167  01433 066060R       LDB MLBX1+1    FIELD
C168  01434 045020R       ADB CCNT
C169  01435 006002       SZB              FULL?
C170  01436 027431R       JMP *-5        NO
C171  01437 127426R       JMP ENOUT,I
C172**
C173***  SPACE FOR A COMMA  **
C174**
C175  01440 000000  EDELM NOP
C176  01441 065020R       LDB CCNT       NO, LOAD CHARACTER COUNT
C177  01442 006003  EDEL1 SZB,RSS        ZERO?
C178  01443 127440R       JMP EDELM,I    YES
C179  01444 045235R       ADB M15        NO, SUBTRACT ZONE WIDTH
C180  01445 006021       SSB,RSS        NEGATIVE RESULT?
C181  01446 027442R       JMP EDEL1      NO
C182  01447 077461R       STB OUTLN      YES, SAVE BLANK COUNT
C183  01450 062141R       LDA .32        FETCH BLANK
```

```
C184  01451 017477R       JSB OUTCR      OUTPUT
C185  01452 037461R       ISZ OUTLN
C186  01453 027450R       JMP *-3          BLANKS
C187  01454 066020R       LDB CCNT       LINE
C188  01455 046251R       ADB M76
C189  01456 006021        SSB,RSS          FULL?
C190  01457 017461R       JSB OUTLN      YES
C191  01460 127440R       JMP EDELM,I
```

```
C193**
C194***  OUTPUT A COMPLETED LINE  **
C195**
C196  01461 000000  OUTLN NOP
C197  01462 062034R       LDA TYPE      FETCH 'CHARACTERS PRINTED' COUNTER
C198  01463 000010       SLA           CORRECT FOR START ON
C199  01464 002004       INA             ODD PRINT POSITION
C200  01465 042020R       ADA CCNT      OUTPUT
C201  01466 066016R       LDB .BUFA       A
C202  01467 017161R       JSB EXECW      LINE
C203  01470 066060R       LDB MLBX1+1   CORRECT
C204  01471 046020R       ADB CCNT
C205  01472 076060R       STB MLBX1+1     MARKER
C206  01473 002400        CLA           RESET COUNT OF
C207  01474 072034R       STA TYPE        CHARACTERS PRINTED
C208  01475 015671R       JSB PRNIN     CLEAN UP
C209  01476 127461R       JMP OUTLN,I
C210**
C211***  ADD A CHARACTER TO OUTPUT BUFFER  **
C212**
C213  01477 000000  OUTCR NOP           CHARACTER IN (A)
C214  01500 073160R       STA IFIX      SAVE CHARACTER
C215  01501 036020R       ISZ CCNT      COUNT IT
C216  01502 066020R       LDB CCNT      FIRST CHARACTER
C217  01503 004010        SLB             OF BUFFER WORD?
C218  01504 036017R       ISZ BADDR     YES, MOVE TO FRESH WORD
C219  01505 162017R       LDA BADDR,I   LOAD BUFFER WORD
C220  01506 004010        SLB           SAVE
C221  01507 001727        ALF,ALF         OTHER
C222  01510 010075        AND M256          CHARACTER
C223  01511 053160R       IOR IFIX      ADD NEW CHARACTER
C224  01512 004010        SLB           POSITION
C225  01513 001727        ALF,ALF         WORD AND
C226  01514 172017R       STA BADDR,I       STORE IT
C227  01515 127477R       JMP OUTCR,I
C228*
C229*
C230  00046         TEMP  EQU TEMPS+1
C231  00047         TEMP1 EQU TEMPS+2
C232  00050         TEMP2 EQU TEMPS+3
C233  00051         TEMP3 EQU TEMPS+4
C234  00052         TEMP4 EQU TEMPS+5
C235  00053         COUNT EQU TEMPS+6
C236  00051         STEMP EQU TEMPS+4
C237  01066         MANT1 EQU SYMCK
C238  01130         MANT2 EQU MDIM
C239  01366         EXPON EQU LETCK
C240  01416         DPFLG EQU BCKSP
C241  01477         ARYAD EQU OUTCR
C242  00402         EOL   EQU CONST
```

```
C244*
C245*****************  BASIC INTERPRETER CONTROL ***************************
C246*
C247*    THIS PROGRAM INTERPRETS THE SYSTEM COMMANDS AND PROVIDES
C248*      I/O  CONTROL FOR THE BASIC INTERPRETER. ALL USER
C249*      COMMUNICATION IS DONE THRU THIS PROGRAM. USER RESPONSES ARE
C250*      CHECKED FOR SYSTEM COMMANDS AND IF A VALID COMMAND IS
C251*      DETECTED  THIS PROGRAM INITIATES APPROPRIATE ACTION.
C252*
C253*
C254*  DATA LOCAL TO MONITOR
C255*
C256  01516 001517R RDYA  DEF READY
C257  01517 003412  READY OCT 3412
C258  01520 051105        ASC 2,READ
      01521 040504
C259  01522 054415        OCT 54415
C260  01523 000216R LFEED DEF LF
C261  01524 000260R QMRKA DEF QMARK
C262  01525 004006R STOPA DEF STCMD
C263  01526 001715R CMNDA DEF CMNDS
C264  01527 000000  LTRAK BSS 1
C265  01530 000021  RC17  DEC 17
C266  01531 000000  ENTRY NOP
C267  01532 060147        LDA 147B        (LWA OF THIS PROG)
C268  01533 002004        INA             FIND START OF FREE CORE
C269  01534 072004C       STA BGNPG       AND SAVE
C270  01535 072005C       STA PG1         AS START OF BUFFER #1
C271  01536 003004        CMA,INA
C272  01537 040100        ADA 100B         (LWAM)
C273  01540 001100        ARS             FREE CORE / 2
C274  01541 006400        CLB
C275  01542 100400        DIV .128        DETERMINE # AVAIL SECTORS
      01543 000172R
C276  01544 002003        SZA,RSS         ZERO SECTORS AVAIL ?
C277  01545 016274R       JSB ERROR       YES, ERROR
C278  01546 100200  NOCOR MPY .128       DETERMINE BUFFER LENGTH
      01547 000172R

PG 000
LN  0279        STA BFRL        AND SAVE
C279  01550 072000        STA BFRL        AND SAVE
C280  01551 042005C       ADA PG1         DETERMINE START ADR OF BUFFER #2
C281  01552 072006C       STA PG2         AND SAVE
C282  01553 016005X       JSB EXEC
C283  01554 001561R       DEF *+5          WORK AREA LIMITS
C284  01555 001530R       DEF RC17
C285  01556 000011C       DEF FTRAK
C286  01557 001527R       DEF LTRAK
C287  01560 015135R       DEF SIZE
C288  01561 002400        CLA
C289  01562 072001C       STA PGPT1
C290  01563 072002C       STA PGPT2
C291  01564 072000C       STA PGPTR
C292*
```

```
C293  01565 062002R FLUSH LDA FWAM
C294  01566 072007C       STA PBUFF      SET PROGRAM BUFFER ADDRESS
C295  01567 072010C       STA PBPTR      SET PROGRAM BUFFER POINTER
C296  01570 062141R       LDA .32        INITIALIZE
C297  01571 072271R       STA BLANK        DELETE CHARACTER FOR GETCR
C298  01572 002400         CLA            SET LINE NUMBER
C299  01573 072033R       STA .LNUM        TO 0 INITIALLY
C300  01574 072015R       STA OLDLN
C301  01575 063741R RDYPT LDA PRNTL        SET TO
C302  01576 073131R       STA LISTR        COMMAND MODE
C303  01577 002400         CLA
C304  01600 073667R       STA DRQST      CLEAR DATA REQUEST FLAG
C305  01601 072023R       STA TFLAG      CLEAR PHOTO READER INPUT FLAG
C306  01602 072024R       STA TTYFL      CLEAR TTY TAPE FLAG
C307  01603 072150R       STA .DFLG      CLEAR DISK INPUT FLAG
C308  01604 072025R       STA ENFLG
C309  01605 073264R       STA XIT    RESET EXECUTION EXIT FLAG
C310  01606 060063         LDA .8
C311  01607 067516R       LDB RDYA
C312  01610 017161R       JSB EXECW
C313  01611 063741R PEXMK LDA PRNTL            SHIFT TO
C314  01612 073131R       STA LISTR        COMMAND MODE
C315  01613 062023R       LDA TFLAG
C316  01614 002002         SZA            IS TAPE FLAG SET?
C317  01615 027744R       JMP PTAPE+1    YES, GET RECORD FROM PHOTO RDR
C318  01616 062150R       LDA .DFLG    IS DISK FLAG SET?
C319  01617 002002         SZA
C320  01620 026052R       JMP NXTRC    GET RECORD FROM DISK
C321  01621 027633R       JMP GTRCD    THIS ADDED (6-APR=71)
```

```
C323  01622 067523R DATAI LDB LFEED     LOAD ADDRESS OF LINE FEED
C324  01623 075040R       STB RSYM      STORE ADDRESS OF READY SYMBOL
C325  01624 062024R       LDA TTYFL     TTY TAPE
C326  01625 002002        SZA             INPUT?
C327  01626 027636R       JMP GTRCD+3     YES, SUPPRESS LINE FEED
C328  01627 003400       CCA           NO
C329  01630 066040R       LDB RSYM      LOAD LF OR '?' ADDRESS
C330  01631 017161R       JSB EXECW   PRINT LF OR '?', NO CR-LF
C331  01632 027636R       JMP *+4
C332  01633 060050  GTRCD LDA M3      PRINT ! TO SAY BASIC
C333  01634 067712R       LDB XPNT    IS READY
C334  01635 017161R       JSB EXECW
C335  01636 062247R       LDA M72
C336  01637 066016R       LDB .BUFA
C337  01640 017124R       JSB EXECR     GET RECORD FROM TTY
C338*
C339  01641 003021  RPRCS CMA,SSA,RSS   SET A=-1-# CHARS AND CHECK FOR
C340  01642 016274R       JSB ERROR     RECORD TOO LONG
C341  01643 072020R RTLE  STA CCNT      -1-# CHARACTERS < 0,SET CCNT
C342  01644 062016R       LDA .BUFA     LOAD BUFFER ADDRESS
C343  01645 000066        CLE,ELA       SHIFT LEFT,LEAST BIT USED AS
C344  01646 072017R       STA BADDR      ODD/EVEN FLAG
C345  01647 017377R       JSB GETCR     FETCH FIRST CHARACTER
C346  01650 027622R       JMP DATAI     NULL RECORD, INPUT AGAIN
C347  01651 067667R       LDB DRQST
C348  01652 006003        SZB,RSS       DATA REQUEST?
C349  01653 027674R       JMP CKRCD     NO DATA REQUEST,GO CHECK RECORD
C350  01654 052167R       CPA S         ASCII S  FIRST CHARACTER?
C351  01655 027760R       JMP STOP      ASSUME STOP REQUESTED
C352  01656 002400        CLA           LINE
C353  01657 017161R       JSB EXECW     FEED
C354  01660 017416R       JSB BCKSP     BACKSPACE
C355  01661 062226R       LDA RMODE     RETURN TO
C356  01662 073131R       STA LISTR       RUN MODE
C357  01663 067667R       LDB DRQST
C358  01664 002400        CLA
C359  01665 073667R       STA DRQST     CLEAR DATA REQUEST FLAG
C360  01666 124001        JMP 1,I       GO TO DATA REQUEST CALLING POINT
C361*
C362*
C363*  THIS SECTION REQUESTS DATA INPUT
C364*
C365  01667 000000  DRQST NOP             EXIT/ENTRY AND FLAG
C366  01670 067741R       LDB PRNTL       SHIFT TO
C367  01671 077131R       STB LISTR       COMMAND MODE
C368  01672 067524R       LDB QMRKA
C369  01673 027623R       JMP DATAI+1   PRINT '?' AND WAIT
```

```
C371*
C372*  THIS SECTION CHECKS RECORD FOR SYSTEM COMMANDS.
C373*
C374  01674 066021R CKRCD LDB SBUFA
C375  01675 076022R       STB SBPTR      INITIALIZE SYNTAX BUFFER POINTER
C376  01676 172022R       STA SBPTR,1    PUT FIRST CHAR IN SYNTAX BUFFER
C377  01677 017366R       JSB LETCK      IS CHARACTER A LETTER
C378  01700 025155R       JMP SYNTX      NO, TRY SYNTAX
C379*
C380  01701 062105R       LDA TBLAD      LOAD SYS CMND TABLE START POINT
C381  01702 064041        LDB M10        LOOK FOR A
C382  01703 016242R       JSB TBSRH      SYSTEM COMMAND
C383  01704 015274R       JSB ERROR      NOT A VALID COMMAND
C384*
C385  01705          INVSC EQU *          INVALID CMND ERROR REFERENCE
C386  01705 060065       LDA .10
C387  01706 042053R       ADA COUNT
C388  01707 043526R       ADA CMNDA      ADD START ADDR. OF CMND ROUTINES
C389  01710 164000        LDB 0,1
C390  01711 124001        JMP 1,I
C391*
C392  01712 001713R XPNT  DEF *+1
C393  01713 020440        ASC 2,! _
      01714 057440
```

```
C395*
C396*  THIS SETS UP AND EXECUTES THE SYSTEM COMMANDS
C397*
C398  01715          CMNDS EQU *         COMMAND LIST REFERENCE
C399*
C400*****************************************
C401*** THE FOLLOWING TABLE ADDED 13-APR-71 ***
C402*****************************************
C403*
C404  01715 001727R        DEF RUN
C405  01716 001730R        DEF SCRTH
C406  01717 001731R        DEF TLIST
C407  01720 001733R        DEF PLIST
C408  01721 001743R        DEF PTAPE
C409  01722 001760R        DEF STOP
C410  01723 001772R        DEF TAPE
C411  01724 001774R        DEF BYEC
C412  01725 001777R        DEF PUNC
C413  01726 002003R        DEF LOAD  p29
C414*
C415*****************************************
C416  01727 027256R RUN    JMP RFASE      GO TO RUN ENTRY POINT
C417*
C418  01730 027565R SCRTH JMP FLUSH      SCRATCH CURRENT PROGRAM
C419  01731 063741R TLIST LDA PRNTL        LIST PROGRAM, TFLAG = 0
C420  01732 006401        CLB,RSS
C421  01733 063740R PLIST LDA PUNCL        PUNCH PROGRAM TFLAG # 0
C422  01734 073131R        STA LISTR      SET DRIVER ADDRESS
C423  01735 006400        CLB
C424  01736 076023R LPUNF STB TFLAG
C425  01737 025641R        JMP LIST      GO TO LIST ENTRY POINT
C426  01740 013242R PUNCL DEF EXECP
C427  01741 013161R PRNTL DEF EXECW
C428  01742 013211R TLPUN DEF LPUN
C429  01743 000000  PTAPE NOP
C430  01744 062025R        LDA ENFLG
C431  01745 002003        SZA,RSS
C432  01746 027751R        JMP *+3
C433  01747 017152R        JSB EXECS
C434  01750 027575R        JMP RDYPT
C435  01751 062163R        LDA .72
C436  01752 066016R        LDB .BUFA
C437  01753 017137R        JSB EXECT    GET RECORD FROM PHOTO READER
C438  01754 002003  PRERR SZA,RSS         YES
C439  01755 027744R        JMP PTAPE+1    NULL RECORD
C440  01756 072023R        STA TFLAG      SET FLAG # 0
C441  01757 027641R        JMP RPRCS      GO PROCESS RECORD
C442*
C443*    STOP COMMAND SERVICE
C444*
C445  01760 067741R STOP  LDB PRNTL      SHIFT TO
C446  01761 077131R        STB LISTR         COMMAND MODE
C447  01762 062263R        LDA MNEG
C448  01763 002006        INA,SZA
C449  01764 027763R        JMP *-1         DELAY FOR 100 MILLISECONDS
```

```
C450  01765 017161R        JSB EXECW     C-R AND L-F
C451  01766 060057         LDA .4
C452  01767 067525R        LDB STOPA
C453  01770 017161R        JSB EXECW     PRINT 'STOP'
C454  01771 027575R        JMP RDYPT
C455*
C456**  SET LINE FEED SUPPRESSION
C457*
C458  01772 072024R TAPE   STA TTYFL      SET TO 'TAPE' MODE
C459  01773 027633R        JMP GTRCD
C460  01774 015005X BYEC   JSB EXEC
C461  01775 001777R        DEF *+2
C462  01776 000061         DEF .6
C463  01777 063742R PUNC   LDA TLPUN
C464  02000 073131R        STA LISTR
C465  02001 005404         CLB,INB
C466  02002 027736R        JMP LPUNF
C467**
C468***     LOAD PROGRAM FROM DISK FILE
C469**
C470  02003 000000  LOAD   NOP
C471  02004 062152R        LDA SPSP      SET FILE NAME
C472  02005 072137R        STA FNAME           TO
C473  02006 072140R        STA FNAME+1           ALL
C474  02007 072141R        STA FNAME+2                BLANKS
C475  02010 062136R        LDA FADR      LOAD FILE NAME ADDRESS
C476  02011 001000         ALS           DOUBLE TO USE AS BYTE POINTER
C477  02012 073160R        STA PNTR
C478  02013 017577R LD1    JSB GETCR     FETCH FILE NAME CHARACTER
C479  02014 025032R        JMP LD2       END OF INPUT FILE, EXIT
C480  02015 072045R        STA .TEMP     SAVE CHARACTER
C481  02016 057160R        LDB PNTR      GET BYTE POINTER
C482  02017 037160R        ISZ PNTR      BUMP POINTER
C483  02020 004065         CLE,ERB       MAKE WORD ADDRESS AND SET
C484  02021 160001         LDA B,I       L/R FLAG THEN GET WORD
C485  02022 002041         SEZ,RSS       (E)=0 FOR LEFT CHAR.
C486  02023 001727         ALF,ALF       POSITION CHARACTER TO LEFT BYTE
C487  02024 010075         AND M256      MASK RIGHT BYTE AND ADD
C488  02025 032045R        IOR .TEMP     NEW CHARACTER
C489  02026 002041         SEZ,RSS       (E)=0 SWAP HALVES
C490  02027 001727         ALF,ALF       OF A
C491  02030 170001         STA B,I       AND SAVE
C492  02031 025013R        JMP LD1       DO IT AGAIN
C493  02032 015005X LD2    JSB EXEC      SEARCH DISK FOR REQUESTED
C494  02033 002037R        DEF *+4       FILE
C495  02034 000151R        DEF .18
C496  02035 002137R        DEF FNAME  p31
C497  02036 002142R        DEF NSECT
C498  02037 062142R        LDA NSECT     IF FILE NOT FOUND
C499  02040 002003         SZA,RSS       PRINT ERROR MESSAGE
C500  02041 016274R        JSB ERROR
C501  02042 062142R NOFIL  LDA NSECT
C502  02043 002004         INA
C503  02044 003004         CMA,INA       MAKE -(SECTORS+1)
C504  02045 072145R        STA ENTLN
```

```
C505  02046 062146R        LDA DNBUF     FORCE READ OF FIRST SECTOR
C506  02047 072144R        STA MLOCN
C507  02050 002400         CLA           SET RELATIVE SECTOR
C508  02051 072143R        STA RLSEC     TO ZERO
C509  02052 062025R NXTRC  LDA ENFLG     END STATEMENT ENCOUNTERED?
C510  02053 002002         SZA
C511  02054 027575R        JMP RDYPT     YES, RETURN TO READY
C512  02055 016061R        JSB DISCR     NO, READ NEXT SOURCE STATEMENT
C513  02056 027575R        JMP RDYPT     END OF FILE, RETURN TO READY
C514  02057 072150R        STA .DFLG     SET DISK INPUT FLAG
C515  02060 027641R        JMP RPRCS     GO PROCESS RECORD
C516**
C517**
C518***     READ A SOURCE STATEMENT FROM THE DISC OUTPUT BUFFER
C519***     WHEN THE BUFFER IS EXHAUSTED ANOTHER SECTOR IS READ
C520***     FROM THE DISK.
C521**
C522**
C523  02061 000000  DISCR NOP
C524  02062 016105R        JSB DRORD      GO GET THE 1ST WORD(= # OF WORDS)
C525  02063 001727         ALF,ALF
C526  02064 010072         AND B77
C527  02065 072153R        STA STLEN     SAVE STATEMENT LENGTH
C528  02066 002003         SZA,RSS       IS IT=0
C529  02067 126061R        JMP DISCR,I   YES, END OF FILE EXIT
C530  02070 003004         CMA,INA
C531  02071 072154R        STA TERMS     SET WORD COUNTER
C532  02072 062016R        LDA .BUFA      SET START
C533  02073 072151R        STA SBUFF      OF 'BUFFER'
C534  02074 016105R DSCR2  JSB DRORD
C535  02075 172151R        STA SBUFF,I
C536  02076 036151R        ISZ SBUFF
C537  02077 036154R        ISZ TERMS      HAVE WE FINISHED THE STATEMENT?
C538  02100 026074R        JMP DSCR2      NO
C539  02101 036061R        ISZ DISCR      YES- RETURN
C540  02102 062153R        LDA STLEN       GET STATEMENT LENGTH
C541  02103 001000         ALS            MAKE CHARACTERS
C542  02104 126061R        JMP DISCR,I
C543**
C544***     READ A WORD FROM DISK INPUT BUFFER
C545***     WHEN THE BUFFER IS EXHAUSTED A NEW SECTOR
C546***     IS READ FROM THE DISK
C547**
C548  02105 000000  DRORD NOP
C549  02106 062144R        LDA MLOCN
C550  02107 052146R        CPA DNBUF      POINTER AT END OF BUFFER
C551  02110 002001         RSS            YES
C552  02111 026120R        JMP DWDS       NO- GET THE NEXT WORD
C553  02112 062147R        LDA DBUFF
C554  02113 072144R        STA MLOCN      SET POINTER TO START OF BUFFER
C555  02114 036145R        ISZ ENTLN      ARE WE AT THE END?
C556  02115 002001         RSS
C557  02116 126061R        JMP DISCR,I    YES, TAKE P+1 EXIT FROM DISCR
C558  02117 016123R        JSB RSECT
C559  02120 162144R DWDS   LDA MLOCN,I    GET NEXT WORD FROM BUFFER
```

```
C560  02121 056144R         ISZ MLOCN       BUMP POINTER
C561  02122 126105R         JMP DRORD,I     EXIT TO DISCR
C562**
C563***     READ A SECTOR FROM DISK FILE
C564**
C565  02123 000000  RSECT  NOP
C566  02124 016005X         JSB EXEC
C567  02125 002134R         DEF *+7       READ NEXT SECTOR
C568  02126 000130R         DEF .14
C569  02127 000002          DEF .2
C570  02130 013457R         DEF FINIS+100      OF
C571  02131 000172R         DEF .128
C572  02132 002137R         DEF FNAME       DISK FILE
C573  02133 002143R         DEF RLSEC
C574  02134 056143R         ISZ RLSEC     INCREMENT RELATIVE SECTOR NUMBER
C575  02135 126123R         JMP RSECT,I
C576*
C577*
C578  01160           PNTR   EQU IFIX        BYTE POINTER
C579  00045           .TEMP  EQU TEMPS       TEMPORARY CHARACTER STORAGE
C580  02136 002137R FADR   DEF *+1          FILE NAME ADDRESS
C581  02137 000000  FNAME  BSS 3            HOLDS FILE NAME
C582  02142 000000  NSECT  BSS 1            NUMBER OF SECTORS IN FILE
C583  02143 000000  RLSEC  BSS 1            RELATIVE FILE SECTOR TO BE READ
C584  02144 000000  MLOCN  BSS 1            SECTOR BUFFER POINTER
C585  02145 000000  ENTLN  BSS 1            SECTOR COUNTER
C586  02146 013657R DNBUF  DEF FINIS+228   END OF SECTOR BUFFER
C587  02147 013457R DBUFF  DEF FINIS+100   START OF SECTOR BUFFER
C588  02150 000000  .DFLG  BSS 1            DISK I/O FLAG
C589  02151 000000  SBUFF  BSS 1             INPUT BUFFER POINTER
C590  02152 020040  SPSP   OCT 20040        (BLANK-BLANK)
C591  02153 000000  STLEN  BSS 1            HOLDS STATEMENT WORD COUNT
C592  02154 000000  TERMS  BSS 1            WORD COUNTER
```

```
C594*
C595* ******************************
C596****                            ***
C597***  CHECK SYNTAX OF STATEMENT  ***
C598****                            ***
C599* ******************************
C600*
C601**
C602***  DETERMINE SEQUENCE NUMBER  **
C603**
C604  02155 066033R SYNTX LDB .LNUM       GET OLD SEQUENCE #
C605  02156 007004        CMB,INB        MAKE NEGATIVE
C606  02157 045015R       ADB OLDLN       IS IT HIGHEST OLD SEQ #
C607  02160 006020        SSB
C608  02161 016223R       JSB SAVIT       YES,   LDB .LNUM
C609  02162 066015R       LDB OLDLN       NO
C610  02165 076015R       STB OLDLN       AND SAVE
C611  02164 016400R       JSB INTCK       RECORD
C612  02165 000256R       DEF MAXSN       SEQUENCE NUMBER
C613  02166 035022R       ISZ SBPTR     SAVE SPACE FOR LENGTH WORD
C614  02167 076033R       STB .LNUM     SAVE LINE NUMBER
C615  02170 066021R       LDB SBUFA     SET
C616  02171 006004        INB             TEMP TO
C617  02172 076046R       STB TEMP          (SBUFF)+1
C618**
C619***  DETERMINE STATEMENT TYPE  **
C620**
C621  02173 050065        CPA .10       NULL STATEMENT?
C622  02174 025466R       JMP DLSTM    YES, DELETE IT
C623  02175 172022R       STA SBPTR,I   NO, RECORD NEXT CHARACTER
C624  02176 062106R       LDA STTYP     PRINT-TABLE ADDRESS
C625  02177 066273R       LDB TBLX    -(NUMBR OF ENTRIES)
C626  02200 016242R       JSB TBSRH     FIND STATEMENT TYPE
C627  02201 016274R       JSB ERROR     NOT FOUND
C628  02202 064042  SYNE1 LDB M9        SET MULTIPLE STORE
C629  02203 076500R       STB MSFLG       TO FALSE
C630  02204 066010C       LDB PBPTR     NULL
C631  02205 056007C       CPB PBUFF       PROGRAM?
C632  02206 002001        RSS           YES
C633  02207 025213R       JMP SYNT1     NO
C634  02210 066002R       LDB FWAM      INSURE NO
C635  02211 076007C       STB PBUFF       SPURIOUS COMMON
C636  02212 076010C       STB PBPTR         EXISTS
C637  02213 066013R SYNT1 LDB ASBTB
C638  02214 075045R       STB TEMPS       SET S-STACK POINTER
C639  02215 006400        CLB           SET DEFINE FLAG
C640  02216 076502R       STB DFLAG       TO FALSE
C641  02217 076504R       STB PFLAG     SET PARAMETER FLAG TO FALSE
C642  02220          SYNT  EQU *
C643  02220 001727        ALF,ALF       COMPUTE
C644  02221 042101R       ADA STBAS         SYNTAX ROUTINE AND
C645  02222 124000        JMP 0,I               BRANCH TO IT
C646  02223 000000  SAVIT  NOP
C647  02224 066033R       LDB .LNUM
C648  02225 036223R       ISZ SAVIT
```

```
C649  02226 126223R       JMP SAVIT,I
C650**
C651***  SINGLE CHARACTER AND/OR FORMULA OPERATORS  **
C652**
C653  02227 000400  QUOTE OCT 400  ::
C654  02230 021040        ASC 1,"
C655  02231 001000  COMMA OCT 1000 ::
C656  02232 026040        ASC 1,,
C657  02233 001400  SMCLN OCT 1400 ::
C658  02234 035440        ASC 1,;
C659  02235 002001  RPARN OCT 2001 ::
C660  02236 024440        ASC 1,)
C661  02237 002401  RBRAC OCT 2401
C662  02240 056440        ASC 1,]
C663  02241 003002  SCMMA OCT 3002 ::
C664  02242 026040        ASC 1,,
C665  02243 003402  ASSOP OCT 3402 ::
C666  02244 056440        ASC 1,=
C667  02245 004007  PLUS  OCT 4007 ::
C668  02246 025440        ASC 1,+
C669  02247 004407  MINUS OCT 4407 ::
C670  02250 026440        ASC 1,-
C671  02251 005010  TIMES OCT 5010 ::
C672  02252 025040        ASC 1,*
C673  02253 005410  DIV   OCT 5410 ::
C674  02254 027440        ASC 1,/
C675  02255 006012  EXPS  OCT 6012 ::
C676  02256 057040        ASC 1,^
C677  02257 006405  GTR   OCT 6405 ::
C678  02260 037040        ASC 1,>
C679  02261 007005  LSS   OCT 7005
C680  02262 056040        ASC 1,<
C681  02263 007405  UNEQL OCT 7405 ::
C682  02264 021440        ASC 1,#
C683  02265 010005  EQUAL OCT 10005 ::
C684  02266 056440        ASC 1,=
C685  02267 010411  UNMIN OCT 10411 ::
C686  02270 026440        ASC 1,-
C687  02271 011020  LBRAC OCT 11020 ::
C688  02272 055440        ASC 1,[
C689  02273 011420  LPARN OCT 11420 ::
C690  02274 024040        ASC 1,(
C691  02275 012011  UPLUS OCT 12011 ::
C692  02276 025440        ASC 1,+
C693  02277 012403  OROP  OCT 12403 ::
C694  02300 000000  MSFLG NOP
C695  02301 013004  ANDOP OCT 13004 ::
C696  02302 000000  DFLAG NOP
C697  02303 013411  NOTOP OCT 13411  ::
C698  02304 000000  PFLAG NOP
C699  02305 014005  GTREQ OCT 14005 ::
C700  02306 000000  UFLAG NOP
C701  02307 014405  LSSEQ OCT 14405
C702*
C703***                    ***
```

```
C704**  LET STATEMENT SYNTAX  **
C705***                      ***
C706*
C707   02310 006400  LETS  CLB  SET "NO STORE" FLAG (B)=0
C708   02311 077534R       STB SFLAG
C709   02312 060043        LDA M8          SET MULTIPLE STORE FLAG
C710   02313 072300R       STA MSFLG         TO TRUE
C711   02314 017120R       JSB FSC         FETCH FORMULA
C712   02315 057534R       CPB SFLAG       DID STORE OCCUR? ( (B)=0 )
C713   02316 016274R       JSB ERROR       NO
C714   02317          SYNE2 EQU *
C715**
C716***     CHECK FOR END OF STATEMENT  **
C717**
C718   02317 050065  EOST  CPA .10         END-OF-STATEMENT?
C719   02320 026476R       JMP ACTST     YES ACCEPT STATEMENT
C720   02321 026071R       JMP NOEOF      NO, ILLEGAL CHARACTER
```

```
C722*
C723***                          ***
C724**  DIM STATEMENT SYNTAX     **
C725***                          ***
C726*
C727  02322 055502R DIMS  ISZ DFLAG      SET DFLAG TO TRUE
C728  02323 017534R        JSB ARRYS      CHECK AN ARRAY
C729  02324 025476R        JMP ACTST     DONE
C730  02325 025323R        JMP DIMS+1     WAS A COMMA, CONTINUE
C731*
C732***                          ***
C733**  COM STATEMENT SYNTAX     **
C734***                          ***
C735*
C736  02326 066010C COMS  LDB PBPTR      HAS PROGRAM BUFFER
C737  02327 055002R        CPB FWAM        BEEN MOVED?
C738  02330 002001        RSS            NO
C739  02331 015274R        JSB ERROR      YES, ILLEGAL COM
C740  02332 076054R SYNE3 STB TEMPS+7    SET ARRAY POINTER
C741  02333 055502R        ISZ DFLAG      SET DEFINE FLAG TO TRUE
C742  02334 003400  COMS1 CCA             SET COMMON FLAG
C743  02335 072304R        STA PFLAG        TO TRUE
C744  02336 017534R        JSB ARRYS      CHECK AN ARRAY
C745  02337 002001        RSS            DONE
C746  02340 025334R        JMP COMS1      MORE ARRAYS
C747  02341 066054R        LDB TEMPS+7    FETCH UPDATED POINTER
C748  02342 076007C        STB PBUFF      SET PROGRAM BUFFER ADDRESS
C749  02343 076010C        STB PBPTR      SET PROGRAM BUFFER POINTER
C750  02344 025476R        JMP ACTST     EXIT
C751*
C752***                          ***
C753**  DEF STATEMENT SYNTAX     **
C754***                          ***
C755*
C756  02345 017645R DEFS  JSB LTR
C757  02346 025356R        JMP SYNE4      FIRST
C758  02347 062047R        LDA TEMP1
C759  02350 001727        ALF,ALF          TWO CHARACTERS
C760  02351 032050R        IOR TEMP2
C761  02352 052257R        CPA FN             'FN'?
C762  02353 002001        RSS            YES
C763  02354 025356R        JMP SYNE4      NO
C001  02355 017645R        JSB LTR        LETTER FOLLOWS?
C002  02356 015274R SYNE4 JSB ERROR      NO
C003  02357 062047R        LDA TEMP1      YES, RECORD A
C004  02360 066156R        LDB .50 **::FUNCTION
C005  02361 017660R        JSB STROP          NAME
C006  02362 062050R        LDA TEMP2      RETRIEVE CHARACTER
C007  02363 017671R        JSB LPCK       LEFT PARENTHESIS?
C008  02364 032263R        IOR FLGBT      YES, SET FORMAL
C009  02365 172022R        STA SBPTR,I      PARAMETER BIT
C010  02366 017562R        JSB VAROP      FETCH SIMPLE VARIABLE
C011  02367 000000        NOP            NONE FOUND
C012  02370 015274R        JSB ERROR      SUBSCRIPTED VARIABLE FOUND
C013  02371 017701R SYNE5 JSB RPCK       RECORD A RIGHT PARENTHESIS
```

```
C014  02372 007400          CCB             ASSIGNMENT
C015  02373 017066R         JSB SYMCK
C016  02374 002242R         DEF ASSOP-1       OPERATOR?
C017  02375 015274R SYNE6   JSB ERROR       NO
C018  02376 000051          LDA M2          YES,
C019  02377 042022R         ADA SBPTR         RETRIEVE
C020  02400 160000          LDA 0,1             PARAMETER
C021  02401 012176R         AND MSK1              AND
C022  02402 072304R         STA PFLAG               SAVE IT
C023  02403 017120R         JSB FSC         FETCH DEFINING FORMULA
C024  02404 025517R         JMP EOST        END-OF-STATEMENT TEST
C025*
C026***                       ***
C027**  REM STATEMENT SYNTAX  **
C028  02405 102501  REMS    LIA 1           IS SW 13 SET?
C029  02406 012210R         AND B20K
C030  02407 002002          SZA
C031  02410 027611R         JMP PEXMK        YES, DELETE THE REM
C032  02411 060065          LDA .10
C033  02412 015437R         JSB CHRST
C034  02413 025476R         JMP ACTST
C035  02414 027611R         JMP PEXMK
C036*
C037***                       ***
C038**  IF STATEMENT SYNTAX   **
C039***                       ***
C040*
C041  02415 017120R IFS     JSB FSC         GET DECISION FORMULA
C042  02416 172022R         STA SBPTR,I     TABLE
C043  02417 062116R         LDA ATHEN         SEARCH
C044  02420 007400          CCB                 FOR
C045  02421 015242R         JSB TBSRH             'THEN'
C046  02422 015274R         JSB ERROR       NOT FOUND
C047  02423           SYNE7 EQU *           GET STATEMENT LABEL NUMBER
C048  02423 036022R         ISZ SBPTR
C049  02424 017377R         JSB GETCR
C050  02425 025070R         JMP EOF
C051  02426 017366R         JSB LETCK     IS NEXT CHAR. A LETTER?
C052  02427 025436R         JMP IFS1      NO ADD GOTO CODE TO BUFFER
C053  02430 172022R         STA SBPTR,I    YES, SAVE CHAR TEMPORARILY
C054  02431 062106R         LDA STTYP      FIND 'CONDITIONAL EXEC'
C055  02432 066273R         LDB TBLX '     STATEMENT TYPE
C056  02433 015242R         JSB TBSRH
C057  02434 015274R         JSB ERROR     NOT FOUND, ERROR!
C058  02435 025220R SYE1    JMP SYNT      FOUND, BRANCH TO SYNTAX ROUTINE
C059  02436 017416R IFS1    JSB BCKSP
C060  02437 062044R         LDA GOTO      REPLACE SEQUENCE NUMBER BY
C061  02440 040047          ADA M4        GO TO SEQUENCE NUMBER
C062  02441 172022R         STA SBPTR,I
C063*
C064***                                     ***
C065**  GO TO AND GOSUB STATEMENT SYNTAX     **
C066***                                     ***
C067  02442 017120R GOTOS   JSB FSC       FETCH AND RECORD FORMULA
C068  02443 025517R         JMP EOST        END-OF-STATEMENT TEST
```

```
C069*
C070*
C071***                          ***
C072**  FOR STATEMENT SYNTAX     **
C073***                          ***
C074*
C075  02444 017562R FORS  JSB VAROP      FETCH SIMPLE VARIABLE
C076  02445 000000        NOP           NONE FOUND
C077  02446 015274R SYNE8 JSB ERROR     SUBSCRIPTED VARIABLE FOUND
C078  02447 007400        CCB
C079  02450 017066R       JSB SYNCK     ASSIGNMENT
C080  02451 002242R       DEF ASSOP-1     OPERATOR?
C081  02452 025375R       JMP SYNE6     NO
C082  02453 017120R       JSB FSC       YES, FETCH INITIAL VALUE FORMULA
C083  02454 172022R       STA SBPTR,I   LOOK
C084  02455 062117R       LDA ATO         FOR
C085  02456 007400        CCB               THE
C086  02457 016242R       JSB TBSRH             'TO'
C087  02460 015274R       JSB ERROR     MISSING
C088  02461 017120R SYNE9 JSB FSC       GET LIMIT FORMULA
C089  02462 050065        CPA .10       END-OF-STATEMENT?
C090  02463 025476R       JMP ACIST     YES
C091  02464 007400        CCB           NO, ERASE
C092  02465 046022R       ADB SBPTR       ZERO
C093  02466 075022R       STB SBPTR         WORD
C094  02467 172022R       STA SBPTR,I   NOW
C095  02470 062120R       LDA ASTEP       LOOK
C096  02471 007400        CCB               FOR
C097  02472 016242R       JSB TBSRH      THE 'STEP'
C098  02473 015274R       JSB ERROR     MISSING
C099  02474 017120R SYE10 JSB FSC       GET STEP SIZE FORMULA
C100  02475 025317R       JMP EOST      END-OF-STATEMENT TEST
C101*
C102***                          ***
C103**  NEXT STATEMENT SYNTAX    **
C104***                          ***
C105*
C106  02476 017562R NXTS  JSB VAROP      FETCH SIMPLE VARIABLE
C107  02477 000000        NOP           NONE FOUND
C108  02500 025446R       JMP SYNE8     SUBSCRIPTED VARIABLE FOUND
C109  02501 025317R       JMP EOST      END-OF-STATEMENT TEST
C110*
C111***                                                  ***
C112**  END, STOP, RESTORE, RETURN STATEMENT SYNTAX       **
C113***                                                  ***
C114*
C115  02502 062023R ENDS  LDA TFLAG
C116  02503 002002        SZA
C117  02504 056025R       ISZ ENFLG
C118  02505 062150R       LDA .DFLG     IS INPUT FROM DISK?
C119  02506 002002        SZA
C120  02507 056025R       ISZ ENFLG     YES, SET FLAG ON END STATEMENT
C121  02510 056022R       ISZ SBPTR
C122  02511 017377R       JSB GETCR     END-OF-STATEMENT?
C123  02512 025476R       JMP ACIST       YES
```

```
C124  02513 025071R        JMP NOEOF      NO
C125*
C126***                    ***
C127**  WAIT STATEMENT SYNTAX  **
C128***                    ***
C129*
C130  02514 017744R WAITS  JSB GETPF
C131  02515 025517R        JMP EOST       END-OF-STATEMENT TEST
C132***
C133**      CALL STATEMENT SYNTAX
C134***
C135  02516 017577R CALLS  JSB GETCR      FETCH AND
C136  02517 025070R        JMP EOF          RECORD
C137  02520 035022R        ISZ SBPTR          LEFT
C138  02521 017671R        JSB LPCK             PARENTHESIS
C139  02522 015562R        JSB PRGIN        FETCH AND RECORD
C140  02523 000077         DEF D100         SUBROUTINE NUMBER
C141  02524 072047R        STA TEMP1
C142  02525 017115R        JSB FNDSB      FIND
C143  02526 160001         LDA 1,I          NUMBER
C144  02527 001727         ALF,ALF            OF
C145  02530 010071         AND .31              PARAMETERS
C146  02531 003000         CMA            RECORD
C147  02532 072054R        STA TEMP5+7      COMPLEMENT - 1
C148  02533 062047R        LDA TEMP1      RETRIEVE CHARACTER
C149  02534 007400  CALL2  CCB
C150  02535 017066R        JSB SYMCK      COMMA?
C151  02536 002230R        DEF COMMA-1
C152  02537 025545R        JMP CALL3      NO
C153  02540 035054R CALL5  ISZ TEMP5+7     YES,MORE
C154  02541 002001         RSS              PARAMETERS PERMITTED?
C155  02542 015274R SYE11  JSB ERROR      NO
C156  02543 017120R        JSB FSC        YES, FETCH
C157  02544 025534R        JMP CALL2        PARAMETER FORMULA
C158  02545 035054R CALL3  ISZ TEMP5+7    ALL PARAMETERS PRESENT?
C159  02546 025542R        JMP SYE11      NO
C160  02547 066272R        LDB PARM
C161  02550 006002         SZB
C162  02551 017701R        JSB RPCK       YES, FETCH RIGHT PARENTHESIS
C163  02552 025517R        JMP EOST       END-OF-STATEMENT TEST
C164*
C165***                    ***
C166**  DATA STATEMENT SYNTAX  **
C167***                    ***
C168*
C169  02553 015402R DATAS  JSB CONST      FETCH A CONSTANT
C170  02554 025426R        JMP SYE12-1    NONE FOUND
C171  02555 017754R        JSB NUMOP      FIX UP PRECEDING OPERATOR
C172  02556 007400         CCB            CHECK
C173  02557 017066R        JSB SYMCK        FOR A
C174  02560 002230R        DEF COMMA-1        COMMA
C175  02561 025517R        JMP EOST       END-OF-STATEMENT TEST
C176  02562 025553R        JMP DATAS      FETCH ANOTHER NUMBER
C177*
C178***                                   ***
```

```
C179**  READ AND INPUT STATEMENT SYNTAX  **
C180***                                  ***
C181*
C182   02563 017562R READS JSB VAROP      RECORD VARIABLE OPERAND
C183   02564 015274R       JSB ERROR     MISSING
C184   02565 000000  SYE13 NOP
C185   02566 007400        CCB           CHECK
C186   02567 017066R       JSB SYMCK       FOR A
C187   02570 002230R       DEF COMMA-1       COMMA
C188   02571 002001        RSS
C189   02572 025563R       JMP READS     IS, FETCH NEXT ITEM
C190   02573 005400        CLB           APPEND
C191   02574 175022R       STB SBPTR,I     END-OF-FORMULA
C192   02575 035022R       ISZ SBPTR         OPERATOR
C193   02576 026317R       JMP EOST      END OF STATEMENT TEST
C194*
C195***                        ***
C196**  PRINT STATEMENT SYNTAX  **
C197***                        ***
C198*
C199   02577 064051  PRIN1 LDB M2          NO,
C200   02600 017066R       JSB SYMCK       COMMA OR
C201   02601 002230R       DEF COMMA-1       SEMICOLON?
C202   02602 025610R       JMP PRIN2     NO
C203   02603 003400  PRIN3 CCA           YES, ENABLE
C204   02604 172046R       STA TEMP,I      FORMULA
C205   02605 035022R       ISZ SBPTR
C206   02606 017377R       JSB GETCR     END-OF-STATEMENT?
C207   02607 025476R       JMP ACTST       YES
C208   02610 007400  PRIN2 CCB
C209   02611 017066R       JSB SYMCK     QUOTE?
C210   02612 002226R       DEF QUOTE-1
C211   02613 025627R       JMP PRIN3     NO
C212   02614 062143R       LDA .34       YES, SET QUOTE AS TERMINATOR
C213   02615 015437R       JSB CHRST       CHARACTER AND FETCH STRING
C214   02616 015274R       JSB ERROR     MISSING QUOTE
C215   02617 062227R SYE14 LDA QUOTE     RECORD
C216   02620 172022R       STA SBPTR,I     QUOTE
C217   02621 035022R       ISZ SBPTR
C218   02622 017377R       JSB GETCR     END-OF-STATEMENT?
C219   02623 025476R       JMP ACTST       YES
C220   02624 007400        CCB           ENABLE
C221   02625 175046R       STB TEMP,I      FORMULA
C222   02626 025577R       JMP PRIN1     NO
C223   02627 135046R PRIN3 ISZ TEMP,I    TAB OR FORMULA PERMITTED?
C224   02630 015274R       JSB ERROR     NO
C225   02631 172022R SYE15 STA SBPTR,I   SEARCH
C226   02632 062122R       LDA ATAB        FOR
C227   02633 007400        CCB               'TAB'
C228   02634 015242R       JSB TBSRH
C229   02635 002401        CLA,RSS       NOT FOUND
C230   02636 062224R       LDA TABCN
C231   02637 007400        CCB           BACKUP
C232   02640 045022R       ADB SBPTR       TO WORD WITH
C233   02641 075022R       STB SBPTR         PREVIOUS OPERATOR
```

```
C234  02642 002003          SZA,RSS        'TAB' ?
C235  02643 026653R         JMP PRIN4      NO
C236  02644 132022R         IOR SBPTR,I
C237  02645 172022R         STA SBPTR,I    YES, RECORD IT
C238  02646 017744R         JSB GETPF      FETCH PARAMETER
C239  02647 006400          CLB            FOLLOW
C240  02650 176022R         STB SBPTR,I      WITH A
C241  02651 036022R         ISZ SBPTR          ZERO
C242  02652 026655R         JMP PRIN5
C243  02653 017416R PRIN4   JSB BCKSP      BACKSPACE OVER LAST CHARACTER
C244  02654 017120R         JSB FSC        FETCH FORMULA
C245  02655 050065  PRIN5   CPA .10        END-OF-STATEMENT?
C246  02656 026476R         JMP ACTST        YES
C247  02657 026577R         JMP PRIN1      NO
C248*
C249***                       ***
C250**  MAT STATEMENT SYNTAX  **
C251***                       ***
C252*
C253  02660 017645R MATS    JSB LTR        FIRST
C254  02661 016274R         JSB ERROR        TWO CHARACTERS
C255  02662 017366R SYE16   JSB LETCK         LETTERS?
C256  02663 026726R         JMP MATS2      NO
C257  02664 036022R         ISZ SBPTR      YES, MOVE TO FRESH S-BUFFER WORD
C258  02665 066047R         LDB TEMP1      RETRIEVE FIRST LETTER AND
C259  02666 005727          BLF,BLF          PUT IT IN THE
C260  02667 030001          IOR 1              UPPER CHARACTER OF (A)
C261  02670 172022R         STA SBPTR,I    SEARCH
C262  02671 062107R         LDA MATIO        FOR
C263  02672 064051          LDB M2             'READ' OR
C264  02673 016242R         JSB TBSRH        'PRINT'
C265  02674 016274R         JSB ERROR      NOT FOUND
C266  02675 052213R SYE17   CPA RDOP       READ?
C267  02676 026714R         JMP MATS1      YES
C268  02677 017550R MATSO   JSB ARRID      RECORD ARRAY
C269  02700 050065          CPA .10        END-OF-STATEMENT?
C270  02701 026476R         JMP ACTST        YES
C271  02702 064051          LDB M2         NO,
C272  02703 017066R         JSB SYMCK        COMMA OR
C273  02704 002230R         DEF COMMA-1        SEMICOLON?
C274  02705 016274R         JSB ERROR      NO
C275  02706 017577R SYE18   JSB GETCR       END-OF-STATEMENT?
C276  02707 026712R         JMP *+3        YES
C277  02710 017416R         JSB BCKSP      NO
C278  02711 026677R         JMP MATSO
C279  02712 036022R         ISZ SBPTR      INCLUDE
C280  02713 026476R         JMP ACTST        PARAMETER
C281  02714 017550R MATS1   JSB ARRID      RECORD ARRAY
C282  02715 017714R         JSB MATSB      IF SUBSCRIPT,
C283  02716 000000          NOP              RECORD IT
C284  02717 050065          CPA .10        END-OF-STATEMENT?
C285  02720 026476R         JMP ACTST        YES
C286  02721 007400          CCB            NO
C287  02722 017066R         JSB SYMCK
C288  02723 002230R         DEF COMMA-1    COMMA?
```

```
C289  02724 026705R        JMP SYE18-1    NO
C290  02725 026714R        JMP MATS1      YES
C291  02726 072050R MATS2  STA TEMP2
C292  02727 062022R        LDA SBPTR      SAVE
C293  02730 073477R        STA ARYAD        OPERAND ADDRESS
C294  02731 062047R        LDA TEMP1      RETRIEVE FIRST LETTER
C295  02732 066152R        LDB .46        RECORD AN
C296  02733 017660R        JSB STROP        ARRAY
C297  02734 062050R        LDA TEMP2      RETRIEVE CHARACTER
C298  02735 007400          CCB            ASSIGNMENT
C299  02736 017066R        JSB SYMCK
C300  02737 002242R        DEF ASSOP-1      OPERATOR?
C301  02740 026575R        JMP SYNE6      NO
C302  02741 163477R        LDA ARYAD,I    YES, RETRIEVE
C303  02742 012176R        AND MSK1         AND SAVE
C304  02743 172046R        STA TEMP,I         PREVIOUS ARRAY IDENTIFIER
C305  02744 017645R        JSB LTR        LETTER NEXT?
C306  02745 027014R        JMP MATS4      NO
C307  02746 017366R        JSB LETCK      YES, SECOND LETTER?
C308  02747 027030R        JMP MATS5      NO
C309  02750 035022R        ISZ SBPTR      YES,
C310  02751 066047R        LDB TEMP1        CONCATENATE
C311  02752 005727          BLF,BLF          LETTERS
C312  02753 030001          IOR 1              AND
C313  02754 172022R        STA SBPTR,I          SEARCH
C314  02755 062112R        LDA MATFN                  FOR
C315  02756 054046          LDB M5                   ARRAY
C316  02757 016242R        JSB TBSRH                      FUNCTION
C317  02760 015274R        JSB ERROR      NOT FOUND
C318  02761 001727  SYE19  ALF,ALF       FOUND
C319  02762 001723          ALF,RAR      POSITION IT,
C320  02763 040062          ADA .7     COMPLETE OPERAND,
C321  02764 007400          CCB              COMBINE
C322  02765 046022R        ADB SBPTR          WITH
C323  02766 130001          IOR 1,I              OPERATOR,
C324  02767 032263R        IOR FLGBT              ADD FLAG BIT,
C325  02770 170001          STA 1,I                AND STORE
C326  02771 012176R        AND MSK1     'INV'
C327  02772 042253R        ADA D200         OR
C328  02773 002021          SSA,RSS          'TRN?
C329  02774 027002R        JMP MATS3    YES
C330  02775 017377R        JSB GETCR    NO, END-OF-STATEMENT?
C331  02776 026476R        JMP ACTST          YES
C332  02777 017714R        JSB MATSB    NO, SUBSCRIPT?
C333  03000 015274R        JSB ERROR    NO
C334  03001 026071R SYE20  JMP NOEOF
C335  03002 017377R MATS3  JSB GETCR
C336  03003 026070R        JMP EOF
C337  03004 017671R        JSB LPCK     GET LEFT PARENTHESIS
C338  03005 017550R        JSB ARRID    FETCH AND RECORD AN ARRAY
C339  03006 017701R        JSB RPCK     RECORD A RIGHT PARENTHESIS
C340  03007 163477R        LDA ARYAD,I  RETRIEVE
C341  03010 012176R        AND MSK1       PREVIOUS ARRAY IDENTIFIER
C342  03011 152046R        CPA TEMP,I   MATCH LEFT-HAND SIDE ARRAY?
C343  03012 015274R        JSB ERROR    YES
```

```
C344  03015 026476R SYE21 JMP ACTST       NO
C345  03014 036022R MATS4 ISZ SBPTR
C346  03015 017671R       JSB LPCK        FETCH LEFT PARENTHESIS
C347  03016 017120R       JSB FSC         FETCH FORMULA
C348  03017 017701R       JSB RPCK        FETCH RIGHT PARENTHESIS
C349  03020 007400       CCB             MULTIPLICATION
C350  03021 017066R       JSB SYMCK         OPERATOR?
C351  03022 002250R       DEF TIMES-1
C352  03023 016274R       JSB ERROR       NO
C353  03024 017550R SYE22 JSB ARRID       YES, FETCH AND RECORD ARRAY
C354  03025 050065       CPA .10         END-OF-STATEMENT?
C355  03026 026476R       JMP ACTST         YES
C356  03027 025071R       JMP NOEOF       NO
C357  03030 072050R MATS5 STA TEMP2
C358  03031 062022R       LDA SBPTR       SAVE
C359  03032 073477R       STA ARYAD         OPERAND ADDRESS
C360  03033 062047R       LDA TEMP1       RETRIEVE
C361  03034 066152R       LDB .46           AND RECORD
C362  03035 017660R       JSB STROP           ARRAY
C363  03036 062050R       LDA TEMP2       END-OF-
C364  03037 050065       CPA .10           STATEMENT?
C365  03040 026476R       JMP ACTST           YES
C366  03041 064050       LDB M3          NO, MUST BE
C367  03042 017066R       JSB SYMCK         A '+',
C368  03043 002244R       DEF PLUS-1          '-', OR '*'
C369  03044 016274R       JSB ERROR       ISN'T
C370  03045 006400 SYE23 CLB             IS, SET FOR FALSE
C371  03046 040065       ADA .8
C372  03047 052251R       CPA TIMES       '*'?
C373  03050 027065R       JMP MATS7       YES
C374  03051 076304R MATS6 STB PFLAG       NO, SET PFLAG
C375  03052 017550R       JSB ARRID       GET SECOND ARRAY
C376  03053 050065       CPA .10         END-OF-STATEMENT?
C377  03054 002001       RSS             YES
C378  03055 025071R       JMP NOEOF       NO
C379  03056 036304R       ISZ PFLAG       WAS OPERATOR A '*'?
C380  03057 026476R       JMP ACTST         NO
C381  03060 163477R       LDA ARYAD,I     YES, RETRIEVE
C382  03061 012176R       AND MSK1          SECOND ARRAY
C383  03062 152046R       CPA TEMP,I      MATCH LEFT-HAND SIDE ARRAY?
C384  03063 016274R SYE24 JSB ERROR       YES
C385  03064 026476R       JMP ACTST         NO
C386  03065 163477R MATS7 LDA ARYAD,I     RETRIEVE
C387  03066 012176R       AND MSK1          ARRAY
C388  03067 007400       CCB             SET FOR TRUE
C389  03070 152046R       CPA TEMP,I      MATCH LEFT-HAND SIDE ARRAY?
C390  03071 027065R       JMP SYE24       YES
C391  03072 027051R       JMP MATS6       NO
C392**
C393***  JUMP TABLE FOR STATEMENT SYNTAX  **
C394**
C395  03073 002310R SYNTB DEF LETS       LET
C396  03074 002322R       DEF DIMS       DIM
C397  03075 002326R       DEF COMS       COM
C398  03076 002345R       DEF DEFS       DEF
```

```
C399  03077 002405R        DEF REMS      REM
C400  03100 002442R        DEF GOTOS     GO TO
C401  03101 002415R        DEF IFS       IF
C402  03102 002444R        DEF FORS      FOR
C403  03103 002476R        DEF NXTS       NEXT
C404  03104 002442R        DEF GOTOS     GOSUB
C405  03105 002510R        DEF ENDS+6      RETURN
C406  03106 002502R        DEF ENDS      END
C407  03107 002510R        DEF ENDS+6      STOP
C408  03110 002514R        DEF WAITS     WAIT
C409  03111 002516R        DEF CALLS     CALL
C410  03112 002553R        DEF DATAS     DATA
C411  03113 002563R        DEF READS     READ
C412  03114 002603R        DEF PRINS     PRINT
C413  03115 002563R        DEF READS     INPUT
C414  03116 002510R        DEF ENDS+6      RESTORE
C415  03117 002560R        DEF MATS      MAT
C416*
C417***                          ***
C418**  FORMULA SYNTAX CHECKER   **
C419***                          ***
C420*
C421  03120 000000  FSC    NOP
C422  03121 002400         CLA           SET LEFT PARENTHESIS
C423  03122 172045R        STA TEMPS,I     COUNT TO ZERO
C424  03123 003400  FSC1   CCA           SET UNARY FLAG
C425  03124 072306R        STA UFLAG       TO TRUE
C426  03125 017562R FSC2   JSB VAROP     LOOK FOR VARIABLE OPERAND
C427  03126 027265R        JMP FSC9      NOT FOUND
C428  03127 027233R        JMP FSC6      SUBSCRIPTED VARIABLE FOUND
C429  03130 017566R        JSB LETCK     FOLLOWED BY LETTER?
C430  03131 027233R        JMP FSC6      NO
C431  03132 064051         LDB M2        YES, LOOK FOR
C432  03133 017526R        JSB MCHCK       'AND' OR 'OR'
C433  03134 062047R        LDA TEMP1     NOT FOUND, FETCH PREVIOUS
C434  03135 001727         ALF,ALF         CHARACTER AND LEFT-JUSTIFY IT
C435  03136 032050R        IOR TEMP2     ADD LATEST CHARACTER
C436  03137 052257R        CPA FN        'FN'?
C437  03140 027166R        JMP FSC4      YES
C438  03141 172022R        STA SBPTR,I   NO,
C439  03142 062111R        LDA PDFNS       SEARCH FOR
C440  03143 066233R        LDB M11            PREDEFINED
C441  03144 016242R        JSB TBSRH                FUNCTION
C442  03145 027152R        JMP FSC3      NOT FOUND
C443  03146 001727         ALF,ALF       ASSEMBLE
C444  03147 001723         ALF,RAR         OPERAND
C445  03150 032265R        IOR FLGBT      ADD FLAG BIT
C446  03151 027174R        JMP FSC5
C447  03152 036306R FSC3   ISZ UFLAG     'NOT' PERMITTED?
C448  03153 027256R        JMP FSC8-1    NO
C449  03154 062121R        LDA ANOT      YES,
C450  03155 007400         CCB             SEARCH FOR
C451  03156 016242R        JSB TBSRH                  'NOT'
C452  03157 027256R        JMP FSC8-1    'NOT' NOT FOUND
C453  03160 007400         CCB           RETRIEVE
```

```
C454  03161  046022R         ADB  SBPTR      PREVIOUS WORD
C455  03162  160001          LDA  1,I          WORD
C456  03163  012220R         AND  OPMSK     SET TO
C457  03164  170001          STA  1,I          NULL OPERAND
C458  03165  027323R         JMP  FSC14
C459  03166  017377R FSC4    JSB  GETCR      IDENTIFYING
C460  03167  026556R         JMP  SYNE4        FUNCTION
C461  03170  017366R         JSB  LETCK          LETTER?
C462  03171  026556R         JMP  SYNE4      NO
C463  03172  040077          ADA  D100       YES,
C464  03173  001723          ALF,RAR           ASSEMBLE AND SAVE
C465  03174  040062 FSC5     ADA  .7 ::
C466  03175  072047R         STA  TEMP1              FUNCTION IDENTIFIER
C467  03176  007400          CCB             RETRIEVE
C468  03177  046022R         ADB  SBPTR        PREVIOUS
C469  03200  160001          LDA  1,I            PROGRAM WORD
C470  03201  012220R         AND  OPMSK      EXTRACT OPERATOR,
C471  03202  032047R         IOR  TEMP1        APPEND OPERAND,
C472  03203  170001          STA  1,I            AND RECORD
C473  03204  017377R         JSB  GETCR      LEFT PARENTHESIS
C474  03205  016274R FSCE1   JSB  ERROR        OR
C475  03206  017671R         JSB  LPCK           LEFT BRACKET?
C476  03207  017357R         JSB  FRCUR      YES, SAVE LOCAL VARIABLES OF FSC
C477  03210  017120R         JSB  FSC        FETCH ACTUAL PARAMETER
C478  03211  017354R         JSB  FPOP       RESTORE LOCAL VARIABLES OF FSC
C479  03212  017701R         JSB  RPCK       FETCH RIGHT PARENTHESIS
C480  03213  027276R         JMP  FSC10+1
C481  03214  064051 FSC7     LDB  M2         CHECK FOR
C482  03215  017066R         JSB  SYMCK        RIGHT PARENTHESIS
C483  03216  002234R         DEF  RPARN-1        OR RIGHT BRACKET
C484  03217  027257R         JMP  FSC8       NOT FOUND
C485  03220  062202R         LDA  B2000 ::
C486  03221  172022R         STA  SBPTR,I      RIGHT PARENTHESIS
C487  03222  062146R         LDA  .41        RESTORE RIGHT PARENTHESIS
C488  03223  007400          CCB             MATCHING
C489  03224  146045R         ADB  TEMPS,I      LEFT
C490  03225  006020          SSB                 PARENTHESIS?
C491  03226  027257R         JMP  FSC8       NO
C492  03227  176045R         STB  TEMPS,I    YES
C493  03230  036022R         ISZ  SBPTR
C494  03231  017377R         JSB  GETCR      FETCH
C495  03232  060065          LDA  .10           CHARACTER
C496  03233  050065 FSC6     CPA  .10        END OF FORMULA?
C497  03234  027257R         JMP  FSC8       YES
C498  03235  072306R         STA  UFLAG      NO, SET UNARY FLAG TO FALSE
C499  03236  064046          LDB  M5         SEARCH FOR A MULTICHARACTER
C500  03237  017326R         JSB  MCBCK        BINARY OPERATOR
C501  03240  162022R         LDA  SBPTR,I    NOT FOUND,
C502  03241  001727          ALF,ALF           RESTORE
C503  03242  010073          AND  B177           CHARACTER
C504  03243  066300R         LDB  MSFLG      SEARCH
C505  03244  017066R         JSB  SYMCK        FOR A
C506  03245  002244R         DEF  PLUS-1         BINARY OPERATOR
C507  03246  002001          RSS             NOT FOUND
C508  03247  027305R         JMP  FSC12      FOUND
```

```
C509  03250 007400          CCB             ASSIGNMENT
C510  03251 017066R         JSB SYMCK
C511  03252 002242R         DEF ASSOP-1       OPERATOR?
C512  03253 027214R         JMP FSC7        NO
C513  03254 073534R         STA SFLAG        YES, SET
C514  03255 027123R         JMP FSC1           'STORE OCCURRED' FLAG
C515  03256 062050R         LDA TEMP2       RETRIEVE LETTER
C516  03257 166045R FSC8    LDB TEMPS,I     ALL LEFT PARENTHESES
C517  03260 006002          SZB               MATCHED?
C518  03261 015274R FSCE2   JSB ERROR        NO
C519  03262 176022R         STB SBPTR,I     YES, RECORD AN
C520  03263 036022R         ISZ SBPTR         END-OF-FORMULA AND
C521  03264 127120R         JMP FSC,I           EXIT WITH CHARACTER IN (A)
C522*
C523  03265 052145R FSC9    CPA .40          LEFT
C524  03266 027301R         JMP FSC11          PARENTHESIS
C525  03267 052170R         CPA B133             OR LEFT BRACKET?
C526  03270 027301R         JMP FSC11        YES
C527  03271 006400          CLB              NO, SET SIGN
C528  03272 076041R         STB SIGN           POSITIVE
C529  03273 015430R         JSB NUMCK        NUMBER?
C530  03274 027310R         JMP FSC13        NO
C531  03275 017754R FSC10   JSB NUMOP        YES, FIX UP PRECEDING OPERATOR
C532  03276 064042          LDB M9           UPDATE
C533  03277 076300R         STB MSFLG          MULTIPLE STORE
C534  03300 027233R         JMP FSC6           FLAG
C535  03301 036022R FSC11   ISZ SBPTR        YES
C536  03302 062207R         LDA B1140 ::
C537  03303 172022R         STA SBPTR,I        IT AND
C538  03304 136045R         ISZ TEMPS,I          COUNT IT
C539  03305 064042  FSC12   LDB M9           UPDATE
C540  03306 076300R         STB MSFLG          MULTIPLE STORE FLAG
C541  03307 027123R         JMP FSC1             FLAG
C542  03310 036306R FSC13   ISZ UFLAG        UNARY OPERATORS PERMITTED?
C543  03311 015274R FSCE3   JSB ERROR        NO
C544  03312 066205R         LDB UNMNC
C545  03313 052147R         CPA .45          '+'?
C546  03314 027320R         JMP *+4          YES
C547  03315 052150R         CPA .45          NO, '-'?
C548  03316 027321R         JMP *+5          YES
C549  03317 027311R         JMP FSCE3        NO
C550  03320 046200R         ADB B1400
C551  03321 036022R         ISZ SBPTR          UNARY
C552  03322 176022R         STB SBPTR,I          OPERATOR
C553  03323 064042  FSC14   LDB M9           UPDATE
C554  03324 076300R         STB MSFLG          MULTIPLE STORE FLAG
C555  03325 027125R         JMP FSC2           FLAG
C556**
C557***  CHECK FOR A MULTICHARACTER BINARY OPERATOR  **
C558***
C559  03326 000000  MCBCK  NOP
C560  03327 172022R         STA SBPTR,I     SEARCH
C561  03330 062110R         LDA MCBOP         FOR 'AND'
C562  03331 016242R         JSB TBSRH                          OR 'OR'
C563  03332 127526R         JMP MCBCK,I     NOT FOUND
```

```
C564  03333 027305R        JMP FSC12      FOUND
C565**
C566***  RESTORE FSC LOCAL QUANTITIES  **
C567**
C568  03334 000000  FPOP   NOP
C569  03335 072047R        STA TEMP1      SAVE CHARACTER
C570  03336 066045R        LDB TEMPS
C571  03337 044046         ADB M5
C572  03340 076045R        STB TEMPS      RESTORE S-STACK TOP
C573  03341 006004         INB
C574  03342 160001         LDA 1,I
C575  03343 072300R        STA MSFLG      RESTORE MULTIPLE STORE FLAG
C576  03344 006004         INB
C577  03345 160001         LDA 1,I
C578  03346 072306R        STA UFLAG      RESTORE UNARY OPERATOR FLAG
C579  03347 006004         INB
C580  03350 160001         LDA 1,I
C581  03351 073120R        STA FSC        RESTORE FSC RETURN ADDRESS
C582  03352 006004         INB
C583  03353 160001         LDA 1,I        RESTORE
C584  03354 073562R        STA VAROP        VAROP RETURN ADDRESS
C585  03355 062047R        LDA TEMP1      RETRIEVE CHARACTER
C586  03356 127334R        JMP FPOP,I
C587**
C588***  SAVE LOCAL QUANTITIES OF FSC  **
C589**
C590  03357 000000  FRCUR  NOP
C591  03360 066045R        LDB TEMPS      FETCH CURRENT S-STACK POINTER
C592  03361 006004         INB            UPDATE IT
C593  03362 062300R        LDA MSFLG      DUMP MULTIPLE STORE
C594  03363 170001         STA 1,I          FLAG ON S-STACK
C595  03364 006004         INB
C596  03365 062306R        LDA UFLAG      STACK UNARY OPERATOR
C597  03366 170001         STA 1,I          FLAG
C598  03367 006004         INB
C599  03370 062120R        LDA FSC        STACK FSC
C600  03371 170001         STA 1,I          RETURN ADDRESS
C601  03372 062562R        LDA VAROP      STACK VAROP RETURN ADDRESS
C602  03373 017375R        JSB SSOV         AND CHECK FOR S-STACK OVERFLOW
C603  03374 127357R        JMP FRCUR,I
C604**
C605***  PUT ITEM ON S-STACK AND CHECK FOR OVERFLOW  **
C606**
C607  03375 000000  SSOV   NOP            STORE QUANTITY
C608  03376 006004         INB            ADVANCE S-STACK POINTER
C609  03377 170001         STA 1,I        SAVE ITEM IN (A)
C610  03400 006004         INB            ADVANCE S-STACK POINTER
C611  03401 076045R        STB TEMPS        AND RECORD IT
C612  03402 007004         CMB,INB
C613  03403 044000R        ADB LWBM       LAST WORD
C614  03404 006020         SSB              EXCEEDED?
C615  03405 015274R FSCE4  JSB ERROR      YES
C616  03406 127375R        JMP SSOV,I
C617**
C618***  CHECK FOR SUBSCRIPT PART  **
```

```
C619**
C620  03407 000000  SBSCK NOP          CHARACTER IN (A)
C621  03410 064051        LDB M2       LEFT BRACKET
C622  03411 017066R       JSB SYMCK      OR
C623  03412 002270R       DEF LBRAC-1      LEFT PARENTHESIS?
C624  03413 127407R       JMP SBSCK,I  NO, RETURN VIA (P+1)
C625  03414 037407R       ISZ SBSCK    YES, SET RETURN TO (P+2)
C626  03415 163477R       LDA ARYAD,I   SET
C627  03416 012244R       AND SBCON
C628  03417 002004        INA                TO
C629  03420 173477R       STA ARYAD,I          SINGLE SUBSCRIPT
C630  03421 062206R       LDA B1100 ::
C631  03422 172022R       STA SBPTR,1    LEFT BRACKET
C632  03423 006400        CLB          DIM OR COM
C633  03424 056502R       CPB DFLAG      STATEMENT?
C634  03425 027477R       JMP SBSC3    NO
C635  03426 016562R       JSB PRGIN              FETCH INTEGER
C636  03427 000075        DEF M256       SUBSCRIPT BOUND
C637  03430 005727        BLF,BLF       SAVE
C638  03431 076047R       STB TEMP1       BOUND
C639  03432 007400        CCB           IS THE
C640  03433 017066R       JSB SYMCK      NEXT CHARACTER
C641  03434 002240R       DEF SCMMA-1      A COMMA?
C642  03435 027442R       JMP SBSC1     NO
C643  03436 137477R       ISZ ARYAD,I   YES, NOTE SECOND SUBSCRIPT
C644  03437 016562R       JSB PRGIN       FETCH SECOND
C645  03440 000075        DEF M256        INTEGER SUBSCRIPT BOUND
C646  03441 002001        RSS
C647  03442 006404  SBSC1 CLB,INB        SET ONE-DIMENSIONAL CASE
C648  03443 056504R       ISZ PFLAG    COM STATEMENT?
C649  03444 027454R       JMP SBSC2    NO
C650  03445 072050R       STA TEMP2     SAVE CHARACTER
C651  03446 050001        LDA 1
C652  03447 052047R       IOR TEMP1     RETRIEVE FIRST BOUND
C653  03450 017130R       JSB MDIM      FIND STORAGE NEED
C654  03451 042054R       ADA TEMPS+7   UPDATE COM
C655  03452 072054R       STA TEMPS+7     STORAGE POINTER
C656  03453 062050R       LDA TEMP2    RETRIEVE NEXT CHARACTER
C657  03454 064051  SBSC2 LDB M2        RIGHT PARENTHESIS
C658  03455 017066R       JSB SYMCK      OR
C659  03456 002254R       DEF RPARN-1      RIGHT BRACKET?
C660  03457 027261R       JMP FSCE2     NO
C661  03460 062203R       LDA B2400 ::
C662  03461 172022R       STA SBPTR,1    RIGHT BRACKET
C663  03462 036022R       ISZ SBPTR    ADJUST S-BUFFER POINTER
C664  03463 017377R       JSB GETCR    FETCH FOLLOWING
C665  03464 000065        LDA .10         CHARACTER
C666  03465 056502R       LDB DFLAG     DIM OR COM
C667  03466 006002        SZB             STATEMENT?
C668  03467 127407R       JMP SBSCK,I   YES
C669  03470 017334R       JSB FPOP     RESTORE FSC LOCAL VARIABLES
C670  03471 064051        LDB M2       RESTORE
C671  03472 046045R       ADB TEMPS      S-STACK
C672  03473 076045R       STB TEMPS        POINTER
C673  03474 006004        INB          FETCH
```

```
C674  03475 164001          LDB 1,I        RETURN ADDRESS
C675  03476 124001          JMP 1,I          AND EXIT
C676  03477 063407R SBSC3   LDA SBSCK      SAVE
C677  03500 066045R         LDB TEMPS       RETURN ADDRESS
C678  03501 017375R         JSB SSOV            ON S-STACK
C679  03502 017357R         JSB FRCUR     SAVE FSC LOCAL VARIABLES
C680  03503 064042          LDB M9        SET MULTIPLE STORE FLAG
C681  03504 075300R         STB MSFLG       TO FALSE
C682  03505 063477R         LDA ARYAD      SAVE
C683  03506 066045R         LDB TEMPS        OPERAND
C684*
C685  03507 017375R         JSB SSOV           ADDRESS
C686  03510 017120R         JSB FSC       GET SUBSCRIPT FORMULA
C687  03511 007400          CCB           CANCEL
C688  03512 046022R         ADB SBPTR       END-OF-FORMULA
C001  03513 075022R         STB SBPTR            OPERATOR
C002  03514 064051          LDB M2         RESTORE
C003  03515 046045R         ADB TEMPS        S-STACK
C004  03516 075045R         STB TEMPS          POINTER
C005  03517 005004          INB            RESTORE
C006  03520 164001          LDB 1,I          OPERAND
C007  03521 077477R         STB ARYAD          ADDRESS
C008  03522 007400          CCB            IS THE
C009  03523 017066R         JSB SYMCK        NEXT CHARACTER
C010  03524 002240R         DEF SCMMA-1        A COMMA?
C011  03525 027454R         JMP SBSC2      NO
C012  03526 137477R         ISZ ARYAD,I    YES, NOTE SECOND SUBSCRIPT
C013  03527 017120R         JSB FSC        GET SUBSCRIPT FORMULA
C014  03530 007400          CCB            CANCEL
C015  03531 046022R         ADB SBPTR       END-OF-FORMULA
C016  03532 075022R         STB SBPTR            OPERATOR
C017  03533 027454R         JMP SBSC2
```

```
C019**
C020***  CHECK SYNTAX OF ARRAY DEFINITIONS  **
C021**
C022  03534 000000  ARRYS NOP
C023  03535 017550R        JSB ARRID      FETCH ARRAY IDENTIFIER
C024  03536 017407R        JSB SBSCK      RECORD A SUBSCRIPT
C025  03537 016274R        JSB ERROR      MISSING SUBSCRIPT
C026  03540 050065  ARRE1 CPA .10         END-OF-STATEMENT?
C027  03541 127534R        JMP ARRYS,I    YES, RETURN VIA (P+1)
C028  03542 007400         CCB            NO,
C029  03543 017066R        JSB SYMCK        MUST BE
C030  03544 002230R        DEF COMMA-1        A COMMA
C031  03545 026071R        JMP NOEOF      ISN'T
C032  03546 037534R        ISZ ARRYS      IS, RETURN
C033  03547 127534R        JMP ARRYS,I      VIA (P+2)
C034**
C035***  FETCH ARRAY IDENTIFIER  **
C036**
C037  03550 000000  ARRID NOP
C038  03551 017645R        JSB LTR      FETCH LETTER
C039  03552 016274R        JSB ERROR    NONE FOUND
C040  03553 062022R ARRE2 LDA SBPTR      SAVE
C041  03554 073477R        STA ARYAD        OPERAND ADDRES
C042  03555 062047R        LDA TEMP1    RECORD
C043  03556 066152R        LDB .46          ARRAY
C044  03557 017660R        JSB STROP         IDENTIFIER
C045  03560 062050R        LDA TEMP2    RETRIEVE FOLLOWING CHARACTER
C046  03561 127550R        JMP ARRID,I
C047**
C048***  CHECK FOR VARIABLE OPERAND  **
C049**
C050  03562 000000  VAROP NOP
C051  03563 017645R        JSB LTR        LETTER?
C052  03564 127562R        JMP VAROP,I    NO, EXIT VIA (P+1)
C053  03565 037562R        ISZ VAROP
C054  03566 052145R        CPA .40        LEFT PARENTHESIS?
C055  03567 027654R        JMP VARO5      YES
C056  03570 052170R        CPA B133       NO, LEFT BRACKET?
C057  03571 027654R        JMP VARO5      YES
C058  03572 037562R        ISZ VAROP      NO
C059  03573 017355R        JSB DIGCK    DIGIT?
C060  03574 027610R        JMP VARO1      NO
C061  03575 062047R        LDA TEMP1    YES, RETRIEVE LETTER,
C062  03576 046154R        ADB .48        AND RESTORE ASCII DIGIT
C063  03577 075047R        STB TEMP1
C064  03600 045242R        ADB D62 ::
C065  03601 006021         SSB,RSS ::
C066  03602 027511R        JMP FSCE3 ::
C067  03603 066047R        LDB TEMP1 ::
C068  03604 017660R        JSB STROP      RECORD VARIABLE
C069  03605 017577R        JSB GETCR    FETCH FOLLOWING
C070  03606 060065         LDA .10          CHARACTER
C071  03607 027614R        JMP VARO2
C072  03610 062047R VARO1 LDA TEMP1     RETRIEVE LETTER,
C073  03611 066153R        LDB .47          SET 'NO DIGIT',
```

```
C074  03612 017660R        JSB STROP        AND RECORD VARIABLE
C075  03613 062050R        LDA TEMP2      RETRIEVE FOLLOWING CHARACTER
C076  03614 072050R VAR02  STA TEMP2      SAVE CHARACTER
C077  03615 006400         CLB            INSIDE A
C078  03616 056304R        CPB PFLAG        DEF STATEMENT?
C079  03617 127562R        JMP VAROP,I    NO, EXIT VIA (P+3)
C080  03620 007400         CCB
C081  03621 046022R        ADB SBPTR        RETRIEVE
C082  03622 160001         LDA 1,I
C083  03623 012176R        AND MSK1           OPERAND
C084  03624 052304R        CPA PFLAG      MATCH PARAMETER?
C085  03625 027630R        JMP VAR04      YES
C086  03626 062050R VAR03  LDA TEMP2      NO, RETRIEVE
C087  03627 127562R        JMP VAROP,I      CHARACTER AND EXIT VIA (P+3)
C088  03630 160001  VAR04  LDA 1,I        SET OPERAND TO
C089  03631 032263R        IOR FLGBT        ACTUAL PARAMETER
C090  03632 170001         STA 1,I            AND RECORD IT
C091  03633 027626R        JMP VAR03
C092  03634 062022R VAR05  LDA SBPTR      SAVE
C093  03635 073477R        STA ARYAD        OPERAND ADDRESS
C094  03636 062047R        LDA TEMP1      RETRIEVE LETTER
C095  03637 066152R        LDB .46        RECORD
C096  03640 017660R        JSB STROP        ARRAY IDENTIFIER
C097  03641 062170R        LDA B133       RETRIEVE LEFT BRACKET
C098  03642 017407R        JSB SBSCK      FETCH SUBSCRIPT
C099  03643 000000         NOP
C100  03644 127562R        JMP VAROP,I    EXIT VIA (P+2)
C101**
C102***  FETCH A LETTER  **
C103**
C104  03645 000000  LTR    NOP
C105  03646 017577R        JSB GETCR
C106  03647 060065         LDA .10
C107  03650 017566R        JSB LETCK      LETTER?
C108  03651 127645R        JMP LTR,I      NO, EXIT VIA (P+1)
C109  03652 037645R        ISZ LTR        YES,
C110  03653 072047R        STA TEMP1        SAVE IT
C111  03654 017577R        JSB GETCR      NEXT CHARACTER
C112  03655 060065         LDA .10          TO (A)
C113  03656 072050R        STA TEMP2      SAVE SECOND CHARACTER
C114  03657 127645R        JMP LTR,I      EXIT VIA (P+2)
C115**
C116***  STORE AN OPERAND NAME  **
C117**
C118  03660 000000  STROP  NOP            LETTER IN (A), NUMBER IN (B)
C119  03661 040077         ADA D100       NUMERICALLY ADJUST THE
C120  03662 046241R        ADB D55          OPERAND NAME
C121  03663 001725         ALF,RAR ::
C122  03664 030001         IOR 1            TWO PARTS
C123  03665 132022R        IOR SBPTR,I    COMPLETE OPERAND-OPERATOR PAIR
C124  03666 172022R        STA SBPTR,I      AND STORE IT
C125  03667 036022R        ISZ SBPTR      UPDATE S-BUFFER POINTER
C126  03670 127660R        JMP STROP,I
```

```
C128**
C129***  CHECK FOR LEFT PARENTHESIS  **
C130**
C131  03671 000000  LPCK  NOP             CHARACTER IN (A)
C132  03672 064051        LDB M2          LEFT PARENTHESIS
C133  03673 017066R       JSB SYMCK         OR
C134  03674 002270R       DEF LBRAC-1         LEFT BRACKET?
C135  03675 027205R       JMP FSCE1       NO
C136  03676 062207R       LDA B1140  ::
C137  03677 172022R       STA SBPTR,I       LEFT PARENTHESIS
C138  03700 127671R       JMP LPCK,I      EXIT
C139**
C140***  CHECK FOR RIGHT PARENTHESIS  **
C141**
C142  03701 000000  RPCK  NOP
C143  03702 064051        LDB M2          RIGHT PARENTHESIS
C144  03703 017066R       JSB SYMCK         OR
C145  03704 002234R       DEF RPARN-1         RIGHT BRACKET?
C146  03705 027261R       JMP FSCE2       NO
C147  03706 062202R       LDA B2000     YES ,RECORD A
C148  03707 172022R       STA SBPTR,I       RIGHT PARENTHESIS
C149  03710 036022R       ISZ SBPTR       UPDATE SYNTAX BUFFER POINTER
C150  03711 017377R       JSB GETCR       FETCH
C151  03712 060065        LDA .10           FOLLOWING CHARACTER
C152  03713 127701R       JMP RPCK,I
C153**
C154***  FETCH MAT STATEMENT SUBSCRIPT  **
C155**
C156  03714 000000  MATSB NOP
C157  03715 064051        LDB M2          LEFT PARENTHESIS
C158  03716 017066R       JSB SYMCK         OR
C159  03717 002270R       DEF LBRAC-1         LEFT BRACKET?
C160  03720 127714R       JMP MATSB,1     NO
C161  03721 037714R       ISZ MATSB       YES, SET RETURN ADDRESS
C162  03722 062206R       LDA B1100     RECORD A
C163  03723 172022R       STA SBPTR,I       LEFT BRACKET
C164  03724 017120R       JSB FSC         FETCH SUBSCRIPT
C165  03725 007400        CCB
C166  03726 017066R       JSB SYMCK       COMMA?
C167  03727 002230R       DEF COMMA-1
C168  03730 002001        RSS             NO
C169  03731 017120R       JSB FSC         YES, FETCH SUBSCRIPT
C170  03732 064051        LDB M2          RIGHT PARENTHESIS
C171  03733 017066R       JSB SYMCK         OR
C172  03734 002234R       DEF RPARN-1         RIGHT BRACKET
C173  03735 027261R       JMP FSCE2
C174  03736 062203R       LDA B2400 ::
C175  03737 172022R       STA SBPTR,I       RIGHT BRACKET
C176  03740 036022R       ISZ SBPTR
C177  03741 017377R       JSB GETCR       END-OF-STATEMENT?
C178  03742 026476R       JMP ACIST         YES
C179  03743 127714R       JMP MATSB,I
```

```
C181**
C182***  FETCH PARENTHESIZED FORMULA  **
C183**
C184  03744 000000  GETPF NOP
C185  03745 017377R       JSB GETCR
C186  03746 026070R       JMP EOF
C187  03747 056022R       ISZ SBPTR
C188  03750 017671R       JSB LPCK       FETCH LEFT PARENTHESIS
C189  03751 017120R       JSB FSC        FETCH FORMULA
C190  03752 017701R       JSB RPCK       GET RIGHT PARENTHESIS
C191  03753 127744R       JMP GETPF,I
C192**
C193***  FLAG OPERATOR WHICH PRECEDES NUMBER  **
C194**
C195  03754 000000  NUMOP NOP
C196  03755 072052R       STA TEMP4
C197  03756 064050        LDB M3         FETCH
C198  03757 046022R       ADB SBPTR        PRECEDING
C199  03760 160001        LDA 1,I            OPERATOR
C200  03761 032265R       IOR FLGBT      ADD FLAG BIT
C201  03762 170001        STA 1,I        REPLACE OPERATOR
C202  03763 062052R       LDA TEMP4
C203  03764 127754R       JMP NUMOP,I
```

```
C205*
C206*    SYSTEM COMMAND TABLE
C207*
C208  03765 000003  SYCMD OCT 00003
C209  03766 051125        ASC 2,RUN      EXECUTE PROGRAM
      03767 047040
C210*
C211  03770 001003        OCT 1003  ::
C212  03771 051503        ASC 2,SCR      SCRATCH PROGRAM
      03772 051040
C213*
C214  03773 001404        OCT 1404  ::
C215  03774 046111        ASC 2,LIST     LIST COMMAND
      03775 051524
C216*
C217  03776 002405        OCT 2405  ::
C218  03777 050114        ASC 5,PLIST    PUNCH LIST COMMAND
      04000 044523
      04001 052040
C219*
C220  04002 005003        OCT 5003  ::
C221  04003 050124        ASC 2,PTA
      04004 040440
C222*
C223  04005 033004        OCT 33004
C224  04006 051524  STCMD ASC 2,STOP     ABORT CURRENT ACTIVITY
      04007 047520
C225*
C226  04010 023003        OCT 23003 ::
C227  04011 052101        ASC 2,TAP      ACTIVATE TTY TAPE MODE
      04012 050040
C228*
C229  04013 024003        OCT 24003 ::
C230  04014 041131        ASC 2,BYE      EXIT SYSTEM
      04015 042440
C231  04016 025005        OCT 26005 ::
C232  04017 050125         ASC 3,PUNCH
      04020 047103
      04021 044040
C233  04022 027404        OCT 27404
C234  04023 046117        ASC 2,LOAD ::
      04024 040504
C235**********************
C236**
C237***  PRINT NAME TABLE FOR OPERATORS  **
C238**
C239  04025 015003  LET   OCT 15003 ::
C240  04026 046105        ASC 2,LET
      04027 052040
C241  04030 015403  DIM   OCT 15403 ::
C242  04031 042111        ASC 2,DIM
      04032 046440
C243  04033 016003  COM   OCT 16003 ::
C244  04034 041517        ASC 2,COM
      04035 046440
```

```
C245  04036 015403  DEF    OCT 16403 ::
C246  04037 042105         ASC 2,DEF
      04040 043040
C247  04041 017003  REM    OCT 17003  ::
C248  04042 051105         ASC 2,REM
      04043 046440
C249  04044 017404  GOTO   OCT 17404  ::
C250  04045 043517         ASC 2,GOTO
      04046 052117
C251  04047 020002  IF     OCT 20002 ::
C252  04050 044506         ASC 1,IF
C253  04051 020403  FOR    OCT 20403 ::
C254  04052 043117         ASC 2,FOR
      04053 051040
C255  04054 021004  NEXT   OCT 21004 ::
C256  04055 047105         ASC 2,NEXT
      04056 054124
C257  04057 021405  GOSUB  OCT 21405  ::
C258  04060 043517         ASC 3,GOSUB
      04061 051525
      04062 041040
C259  04063 022006  RTRN   OCT 22006  ::
C260  04064 051105         ASC 3,RETURN
      04065 052125
      04066 051116
C261  04067 022403  END    OCT 22403 ::
C262  04070 042516         ASC 2,END
      04071 042040
C263  04072 023004  STP    OCT 23004 ::
C264  04073 051524         ASC 2,STOP
      04074 047520
C265  04075 023404  WAIT   OCT 23404 ::
C266  04076 053501         ASC 2,WAIT
      04077 044524
C267  04100 024004  CALL   OCT 24004 ::
C268  04101 041501         ASC 2,CALL
      04102 046114
C269  04103 024404  DATA   OCT 24404  ::
C270  04104 042101         ASC 2,DATA
      04105 052101
C271  04106 025004  READ   OCT 25004 ::
C272  04107 051105         ASC 2,READ
      04110 040504
C273  04111 025405  PRINT  OCT 25405 ::
C274  04112 050122         ASC 3,PRINT
      04113 044516
      04114 052040
C275  04115 026005  INPUT  OCT 26005 ::
C276  04116 044516         ASC 3,INPUT
      04117 050125
      04120 052040
C277  04121 026407  RSTOR  OCT 26407 ::
C278  04122 051105         ASC 4,RESTORE
      04123 051524
      04124 047522
```

```
      04125 042440
C279  04126 027003  MAT   OCT 27003 ::
C280  04127 046501        ASC 2,MAT
      04130 052040
C281  04131 027404  THEN  OCT 27404
C282  04132 052110        ASC 2,THEN
      04133 042516
C283  04134 030002  TO    OCT 30002  ::
C284  04135 052117        ASC 1,TO
C285  04136 030404  STEP  OCT 30404 ::
C286  04137 051524        ASC 2,STEP
      04140 042520
C287  04141 013403  NOT   OCT 13403
C288  04142 047117        ASC 2,NOT
      04143 052040
C289  04144 013003  AND   OCT 13003
C290  04145 040516        ASC 2,AND
      04146 042040
C291  04147 012402  OR    OCT 12402
C292  04150 047522        ASC 1,OR
C293  04151 014002  GTE   OCT 14002 ::
C294  04152 037075        ASC 1,>=
C295  04153 014402  LTE   OCT 14402 ::
C296  04154 036075        ASC 1,<=
C297  04155 074002  AUNEW OCT 74002 ::
C298  04156 036076        ASC 1,<>
C299*
C300  04157 000403  TAB   OCT 403 ::
C301  04160 052101        ASC 2,TAB
      04161 041040
C302  04162 001003  SIN   OCT 1003 ::
C303  04163 051511        ASC 2,SIN
      04164 047040
C304  04165 001403  COS   OCT 1403
C305  04166 041517        ASC 2,COS
      04167 051440
C306  04170 002003  TAN   OCT 2003 ::
C307  04171 052101        ASC 2,TAN
      04172 047040
C308  04173 002403  ATN   OCT 2403 ::
C309  04174 040524        ASC 2,ATN
      04175 047040
C310  04176 003003  EXPN  OCT 3003 ::
C311  04177 042530        ASC 2,EXP
      04200 050040
C312  04201 003403  LOG   OCT 3403 ::
C313  04202 046117        ASC 2,LOG
      04203 043440
C314  04204 004003  ABS   OCT 4003 ::
C315  04205 040502        ASC 2,ABS
      04206 051440
C316  04207 004403  SQR   OCT 4403 ::
C317  04210 051521        ASC 2,SQR
      04211 051040
C318  04212 005003  INT   OCT 5003 ::
```

```
C319  04213 044516          ASC 2,INT
      04214 052040
C320  04215 005403  RND     OCT 5403 ::
C321  04216 051116          ASC 2,RND
      04217 042040
C322  04220 006003  SGN     OCT 6003 ::
C323  04221 051507          ASC 2,SGN
      04222 047040
C324  04223 006403  ZER     OCT 6403 ::
C325  04224 055105          ASC 2,ZER
      04225 051040
C326  04226 007003  CON     OCT 7003 ::
C327  04227 041517          ASC 2,CON
      04230 047040
C328  04231 007403  IDN     OCT 7403 ::
C329  04232 044504          ASC 2,IDN
      04233 047040
C330  04234 010003  INV     OCT 10003 ::
C331  04235 044516          ASC 2,INV
      04236 053040
C332  04237 010403  TRN     OCT 10403 ::
C333  04240 052122          ASC 2,TRN
      04241 047040
C334**
C335***  TABLE SEARCH FOR MULTICHARACTER SYMBOLS  **
C336**
C337  04242 000000  TBSRH  NOP
C338  04243 072362R        STA TABLE      STORE TABLE ADDRESS
C339  04244 076055R        STB LNGTH      STORE -(NUMBER OF ENTRIES)
C340  04245 062017R        LDA BADDR      SAVE
C341  04246 072051R        STA TEMP3        INPUT
C342  04247 062020R        LDA CCNT           BUFFER
C343  04250 072052R        STA TEMP4            STATUS
C344  04251 062022R        LDA SBPTR      INITIALIZE END-OF-SYMBOL
C345  04252 072400R        STA SMEND        POINTER
C346  04253 002404          CLA,INA        COUNT FIRST CHARACTER OF
C347  04254 072626R        STA SLENG        SYMBOL
C348  04255 162022R        LDA SBPTR,I    FETCH PARTIAL SYMBOL
C349  04256 010073          AND B177       TWO
C350  04257 152022R        CPA SBPTR,I      CHARACTERS?
C351  04260 002001          RSS            NO
C352  04261 026304R        JMP TSR10      YES
C353  04262 001727          ALF,ALF        LEFT-JUSTIFY
C354  04263 032141R        IOR .32          FIRST CHARACTER AND
C355  04264 172022R        STA SBPTR,I        APPEND BLANK
C356  04265 017377R TSRC1  JSB GETCR      FETCH NEXT CHARACTER
C357  04266 026355R        JMP TSRC9      END-OF-STATEMENT
C358  04267 066626R        LDB SLENG      CHECK FOR
C359  04270 054065          CPB .10        IMPOSSIBLE LENGTH
C360  04271 026355R        JMP TSRC9
C361  04272 004010          SLB            EVEN-NUMBERED CHARACTER?
C362  04273 026301R        JMP TSRC2      YES
C363  04274 036400R        ISZ SMEND      NO, FETCH FRESH WORD,
C364  04275 001727          ALF,ALF          LEFT-JUSTIFY CHARACTER,
C365  04276 032141R        IOR .32            APPEND BLANK,
```

```
C366  04277 172400R        STA SMEND,I          AND STORE
C367  04300 026504R        JMP TSR10
C368  04301 042240R TSRC2  ADA M32          DELETE BLANK,
C369  04302 142400R        ADA SMEND,I       FILL SECOND CHARACTER,
C370  04303 172400R        STA SMEND,I         AND STORE
C371  04304 036626R TSR10  ISZ SLENG         COUNT IT
C372  04305 066055R        LDB LNGTH         INITIALIZE TABLE LENGTH
C373  04306 076053R        STB COUNT           COUNTER
C374  04307 062562R        LDA TABLE
C375  04310 072553R TSRC3  STA TBLPT         SET TABLE POINTER
C376  04311 162553R        LDA TBLPT,I       EXTRACT SYMBOL LENGTH
C377  04312 010070         AND .15             FROM TABLE AND COMPARE
C378  04313 052626R        CPA SLENG             WITH CURRENT SYMBOL
C379  04314 026323R        JMP TSRC5         EQUAL?
C380  04315 040056  TSRC4  ADA .3            DIFFERENT,
C381  04316 001100         ARS                 UPDATE
C382  04317 042553R        ADA TBLPT             TABLE POINTER
C383  04320 036053R        ISZ COUNT         MORE ENTRIES?
C384  04321 026310R        JMP TSRC3         YES
C385  04322 026265R        JMP TSRC1         NO
C386  04323 066553R TSRC5  LDB TBLPT         SET POINTER TO
C387  04324 076603R        STB TSPTR           TABLE SYMBOL
C388  04325 066022R        LDB SBPTR         SET (B) TO INPUT
C389  04326 026332R        JMP TSRC7           SYMBOL POINTER
C390  04327 056400R TSRC6  CPB SMEND         ALL OF SYMBOL CONSIDERED?
C391  04330 026340R        JMP TSRC8         YES, MATCH OCCURRED
C392  04331 006004         INB               NO, INCREMENT
C393  04332 036603R TSRC7  ISZ TSPTR           SYMBOL POINTERS
C394  04333 162603R        LDA TSPTR,I       FETCH WORD FROM TABLE
C395  04334 150001         CPA 1,1           MATCH WITH INPUT SYMBOL?
C396  04335 026327R        JMP TSRC6         YES
C397  04336 062626R        LDA SLENG         NO, WRONG
C398  04337 026515R        JMP TSRC4           SYMBOL
C399  04340 162553R TSRC8  LDA TBLPT,I       EXTRACT
C400  04341 012220R        AND UPMSK           SYMBOL CODE
C401  04342 172022R        STA SBPTR,I
C402  04343 162553R        LDA TBLPT,I
C403  04344 001727         ALF,ALF
C404  04345 001222         RAL,RAL   ::
C405  04346 001200         RAL       ::
C406  04347 010062         AND .7
C407  04350 003000         CMA
C408  04351 072272R        STA PARM
C409  04352 162022R        LDA SBPTR,I
C410  04353 036242R        ISZ TBSRH             AND RETURN VIA
C411  04354 126242R        JMP TBSRH,I             'SUCCESS' EXIT
C412  04355 062051R TSRC9  LDA TEMP3         RESTORE
C413  04356 072017R        STA BADDR           INPUT
C414  04357 062052R        LDA TEMP4             BUFFER
C415  04360 072020R        STA CCNT                STATUS
C416  04361 126242R        JMP TBSRH,I       'FAILURE' EXIT
C417**
C418***  FETCH AND RECORD PROGRAM INTEGER  **
C419**
C420  04362 000000  PRGIN NOP
```

```
C421  04363 162022R        LDA SBPTR,I   SET
C422  04364 032263R        IOR FLGBT       'INTEGER
C423  04365 040056         ADA .3             FOLLOWS'
C424  04366 172022R        STA SBPTR,I          OPERAND
C425  04367 162362R        LDA PRGIN,I   GIVE ADDRESS
C426  04370 072375R        STA PRGI1       TO INTCK
C427  04371 036022R        ISZ SBPTR
C428  04372 017377R        JSB GETCR
C429  04373 016274R SYE25  JSB ERROR
C430  04374 016400R        JSB INTCK     FETCH
C431  04375 000000  PRGI1  NOP
C432  04376 036362R        ISZ PRGIN
C433  04377 126362R        JMP PRGIN,I
C434**
C435***  BUILD AN INTEGER  **
C436**
C437  04400 000000  INTCK  NOP           CHARACTER IN (A)
C438  04401 006400         CLB           STORE
C439  04402 076626R        STB INTGR       PARTIAL RESULT
C440  04403 017353R INTC1  JSB DIGCK     DIGIT?
C441  04404 026422R        JMP INTC2     NO
C442  04405 103101         CLO
C443  04406 066626R        LDB INTGR     MULTIPLY
C444  04407 044001         ADB 1           PARTIAL
C445  04410 044001         ADB 1             RESULT
C446  04411 046626R        ADB INTGR           BY
C447  04412 044001         ADB 1                 10
C448  04413 044000         ADB 0         ADD LATEST DIGIT
C449  04414 102201         SOC           OVERFLOW?
C450  04415 026373R        JMP SYE25     YES
C451  04416 076626R        STB INTGR     STORE PARTIAL RESULT
C452  04417 017377R        JSB GETCR     NO, FETCH
C453  04420 060065         LDA .10         NEXT CHARACTER
C454  04421 026403R        JMP INTC1
C455  04422 066626R INTC2  LDB INTGR     ZERO
C456  04423 006003         SZB,RSS         INTEGER?
C457  04424 026373R        JMP SYE25     YES
C458  04425 176022R        STB SBPTR,I   NO, RECORD IT
C459  04426 166400R        LDB INTCK,I   INTEGER
C460  04427 164001         LDB 1,I         TOO
C461  04430 046626R        ADB INTGR         LARGE?
C462  04431 006021         SSB,RSS
C463  04432 026373R        JMP SYE25     YES
C464  04433 066626R        LDB INTGR     NO,
C465  04434 036022R        ISZ SBPTR       RETURN WITH
C466  04435 036400R        ISZ INTCK         INTEGER
C467  04436 126400R        JMP INTCK,I         IN (B)
C468**
C469***  PROCESS CHARACTER STRING  **
C470**
C471  04437 000000  CHRST  NOP
C472  04440 072050R        STA TEMP2     RECORD TERMINATOR CHARACTER
C473  04441 060065         LDA .10       DUMMY
C474  04442 072271R        STA BLANK       DELETE CHARACTER
C475  04443 017377R CHRS1  JSB GETCR
```

```
C476  04444  026462R        JMP CHRS3      TO END-OF-STATEMENT EXIT
C477  04445  052050R        CPA TEMP2      TERMINATOR CHARACTER?
C478  04446  026461R        JMP CHRS2      YES
C479  04447  132022R        IOR SBPTR,I    NO, FILL
C480  04450  172022R        STA SBPTR,I      SECOND CHARACTER
C481  04451  017377R        JSB GETCR
C482  04452  026462R        JMP CHRS3      TO END-OF-STATEMENT EXIT
C483  04453  052050R        CPA TEMP2      TERMINATOR CHARACTER?
C484  04454  026461R        JMP CHRS2      YES
C485  04455  036022R        ISZ SBPTR      NO, MOVE TO NEW WORD
C486  04456  001727         ALF,ALF          AND STORE
C487  04457  172022R        STA SBPTR,I        FIRST CHARACTER
C488  04460  026443R        JMP CHRS1
C489  04461  036437R CHRS2  ISZ CHRST      SET (P+2) EXIT
C490  04462  036022R CHRS3  ISZ SBPTR      MOVE TO NEXT BUFFER WORD
C491  04463  062141R        LDA .32        RESTORE BLANK AS
C492  04464  072271R        STA BLANK        DELETE CHARACTER
C493  04465  126437R        JMP CHRST,I
```

```
C495**
C496***  DELETE STATEMENT  **
C497**
C498  04466 162021R DLSTM LDA SBUFA,I   LOAD SEQUENCE NUMBER
C499  04467 016553R       JSB FNDPS     FIND STATEMENT TO BE DELETED
C500  04470 027611R       JMP PEXMK       DOESN'T
C501  04471 027611R       JMP PEXMK       EXIST
C502  04472 002400        CLA           ZERO WORD SKIP FOR .DESTINATION
C503  04473 006004        INB           ADDRESS OF SOURCE WORD SKIP IN B
C504  04474 016603R       JSB CLPRG     CLOSE UP PROGRAM
C505  04475 027611R       JMP PEXMK       EXIT TO PHASE 1 WAIT
C506*
C507***                  ***
C508**  ACCEPT STATEMENT  **
C509***                  ***
C510*
C511  04476 062021R ACTST LDA SBUFA     COMPUTE
C512  04477 003004        CMA,INA         LENGTH
C513  04500 042022R       ADA SBPTR         OF STATEMENT
C514  04501 172046R       STA TEMP,I          AND RECORD IT
C515  04502 066010C       LDB PBPTR       PICK UP STATEMENT ADDR.
C516  04503 076051R       STB TEMP3       AND SAVE
C517  04504 162021R       LDA SBUFA,I   LOAD SEQUENCE NUMBER
C518  04505 003004        CMA,INA         MAKE SEQ # NEGATIVE
C519  04506 042015R       ADA OLDLN       ADD PREVIOUS SEQ #
C520  04507 002020        SSA             P-SEQ < S-SEQ ?
C521  04510 026531R       JMP ACCS1       YES, APPEND STATEMENT TO PROG.
C522  04511 162021R       LDA SBUFA,I     NO, CONTINUE CHECKING
C523  04512 016553R       JSB FNDPS     SEARCH ON SEQUENCE NUMBER
C524  04513 026531R       JMP ACCS1     APPEND STATEMENT TO PROGRAM
C525  04514 026547R       JMP ACCS4     INSERT STATEMENT IN PROGRAM
C526  04515 006004        INB           REPLACE STATEMENT IN PROGRAM
C527  04516 060001        LDA 1
C528  04517 017204R       JSB DREAD     COMPARE LENGTHS OF
C529  04520 003004        CMA,INA         STATEMENT BEING REPLACED
C530  04521 142046R       ADA TEMP,I        AND STATEMENT
C531  04522 002003        SZA,RSS             REPLACING IT
C532  04523 026533R       JMP ACCS2     EQUAL
C533  04524 002021        SSA,RSS
C534  04525 026550R       JMP ACCS4+1   SHORTER
C535  04526 162046R       LDA TEMP,I    LONGER,
C536  04527 016603R       JSB CLPRG       CLOSE UP PROGRAM
C537  04530 026533R       JMP ACCS2
C538  04531 162046R ACCS1 LDA TEMP,I    LOAD PROGRAM SPACE REQUIREMENT
C539  04532 016626R       JSB OVCHK     SUFFICIENT PROGRAM SPACE LEFT?
C540  04533 006400  ACCS2 CLB           YES, SET COUNTER TO ZERO
C541  04534 062021R       LDA SBUFA     INITIALIZE
C542  04535 072050R       STA TEMP2       SOURCE ADDRESS
C543  04536 162050R ACCS3 LDA TEMP2,I   TRANSFER WORD FROM
C544  04537 017151R       JSB DSTOR     S-BUFFER TO PROGRAM SPACE
C545  04540 000051R       DEF TEMP3
C546  04541 036050R       ISZ TEMP2     INCREMENT SOURCE AND
C547  04542 036051R       ISZ TEMP3       .DESTINATION ADDRESSES
C548  04543 006004        INB           BUMP COUNTER
C549  04544 156046R       CPB TEMP,I    ENTIRE STATEMENT MOVED?
```

```
C550   04545 027611R        JMP PEXMK       YES
C551   04546 026536R        JMP ACCS3     NO
C552   04547 162046R ACCS4 LDA TEMP,I     LOAD PROGRAM SPACE REQUIREMENT
C553   04550 016626R        JSB OVCHK     SUFFICIENT PROGRAM SPACE LEFT?
C554   04551 016353R        JSB MVTOH     MAKE
C555   04552 026533R        JMP ACCS2       ROOM
```

```
C557**
C558***  FIND SEQUENTIAL POSITION  **
C559**
C560  04553 000000  FNDPS NOP
C561  04554 072051R       STA TEMP3     SAVE SEQUENCE NUMBER
C562  04555 066007C       LDB PBUFF     STARTING ADDRESS
C563  04556 056010C FNDP1 CPB PBPTR     END OF PROGRAM?
C564  04557 026601R       JMP FNDP4     YES, EXIT VIA (P+1)
C565  04560 060001        LDA 1
C566  04561 017204R       JSB DREAD       SUBTRACT PROGRAM
C567  04562 003004        CMA,INA         SEQUENCE NUMBER FROM
C568  04563 042051R       ADA TEMP3       S-BUFFER SEQUENCE NUMBER
C569  04564 002003        SZA,RSS       EQUAL?
C570  04565 026577R       JMP FNDP2     YES, SET EXIT TO (P+3)
C571  04566 002020        SSA           NO, P-SEQ NO > S-SEQ NO ?
C572  04567 026600R       JMP FNDP3     YES, SET EXIT TO (P+2)
C573  04570 060001        LDA 1         POINT (A) TO
C574  04571 002004        INA             PROGRAM ADDRESS INCREMENT
C575  04572 073130R       STA TMP2
C576  04573 017204R       JSB DREAD
C577  04574 044000        ADB 0           COMPUTE NEW ADDRESS
C578  04575 063130R       LDA TMP2
C579  04576 026556R       JMP FNDP1
C580  04577 036553R FNDP2 ISZ FNDPS
C581  04600 036553R FNDP3 ISZ FNDPS
C582  04601 075051R FNDP4 STB TEMP3     SAVE STATEMENT ADDRESS
C583  04602 126553R       JMP FNDPS,I
C584**
C585***  DELETE SPACE IN PROGRAM  **
C586**
C587  04603 000000  CLPRG NOP           REFERENCE LOCATION IN TEMP3
C588  04604 042051R       ADA TEMP3     SKIP (A) LOCATIONS FROM TEMP3
C589  04605 072052R       STA TEMP4       AND SAVE .DESTINATION ADDRESS
C590  04606 060001        LDA 1
C591  04607 017204R       JSB DREAD      SKIP TO END OF STATEMENT BEING
C592  04610 070001        STA 1
C593  04611 046051R       ADB TEMP3       DELETED, SOURCE ADDRESS IN (B)
C594  04612 056010C CLPR1 CPB PBPTR     ALL OF PROGRAM MOVED?
C595  04613 026623R       JMP CLPR2     YES
C596  04614 060001        LDA 1
C597  04615 017204R       JSB DREAD      NO, MOVE WORD FROM SOURCE TO
C598  04616 017151R       JSB DSTOR      .DESTINATION ADDRESS
C599  04617 000052R       DEF TEMP4
C600  04620 036052R       ISZ TEMP4     INCREMENT .DESTINATION ADDRESS
C601  04621 006004        INB           INCREMENT SOURCE ADDRESS
C602  04622 026612R       JMP CLPR1
C603  04623 062052R CLPR2 LDA TEMP4     SET END-OF-PROGRAM
C604  04624 072010C       STA PBPTR       POINTER
C605  04625 126603R       JMP CLPRG,I
C606**
C607***  CHECK FOR PROGRAM SPACE OVERFLOW  **
C608**
C609  04626 000000  OVCHK NOP           NEW WORD REQUIREMENT IN (A)
C610  04627 066010C       LDB PBPTR     SET SOURCE ADDRESS
C611  04630 076050R       STB TEMP2       FOR PROGRAM RELOCATION
```

```
C612  04631  103101        CLO
C613  04632  044000        ADB 0          SET .DESTINATION
C614  04633  076052R       STB TEMP4        ADDRESS
C615  04634  102201        SOC            ENOUGH FREE SPACE ?
C616  04635  027405R       JMP FSCE4        NO, PROGRAM SPACE OVERFLOW
C617  04636  066052R       LDB TEMP4        YES, RELOCATE FREE
C618  04637  076010C       STB PBPTR          PROGRAM SPACE POINTER
C619  04640  126626R       JMP UVCHK,I
```

```
C621*
C622* **********************
C623****                    ***
C624***  LIST THE PROGRAM   ***
C625****                    ***
C626* **********************
C627*
C628  04641 062010C LIST  LDA PBPTR #
C629  04642 072001R       STA LWBM+1
C630  04643 072272R       STA PARM
C631  04644 065007C       LDB PBUFF       INITIALIZE TO FIRST
C632  04645 076045R       STB TEMPS         STATEMENT OF PROGRAM
C633  04646 017377R       JSB GETCR       SEQUENCE NUMBER GIVEN?
C634  04647 025707R       JMP LIST0       NO
C635  04650 066016R       LDB .BUFA       YES, SET FOR
C636  04651 076022R       STB SBPTR         SEQUENCE NUMBER
C637  04652 016400R       JSB INTCK         FETCH
C638  04653 000256R       DEF MAXSN         IT
C639  04654 072045R       STA TEMPS
C640  04655 162016R       LDA .BUFA,I     LOAD SEQUENCE NUMBER
C641  04656 016553R       JSB FNDPS       FIND INTIAL STATEMENT
C642  04657 027575R       JMP RDYPT
C643  04660 000000        NOP             SAVE
C644  04661 062045R       LDA TEMPS
C645  04662 076045R       STB TEMPS
C646  04663 052151R       CPA .44
C647  04664 017377R       JSB GETCR
C648  04665 025707R       JMP LIST0
C649  04666 066016R       LDB .BUFA
C650  04667 076022R       STB SBPTR
C651  04670 016400R       JSB INTCK
C652  04671 000256R       DEF MAXSN
C653  04672 162016R       LDA .BUFA,I
C654  04673 016553R       JSB FNDPS
C655  04674 025707R       JMP LIST0
C656  04675 025707R       JMP LIST0
C657  04676 060001        LDA 1
C658  04677 006004        INB
C659  04700 073130R       STA TMP2
C660  04701 060001        LDA 1
C661  04702 017204R       JSB DREAD
C662  04703 064000        LDB 0
C663  04704 063130R       LDA TMP2
C664  04705 040001        ADA 1
C665  04706 072001R       STA LWBM+1
C666  04707 006400  LIST0 CLB             HIGH-SPEED
C667  04710 056023R       CPB TFLAG         PUNCH?
C668  04711 025714R       JMP LIST1       NO
C669  04712 062170R       LDA B133        YES, EMIT
C670  04713 117131R       JSB LISTR,I       LEADER
C671  04714 065045R LIST1 LDB TEMPS       MORE
C672  04715 056001R       CPB LWBM+1
C673  04716 027117R       JMP LIS13       NO
C674  04717 003400        CCA             INITIALIZE
C675  04720 042021R       ADA SBUFA         OUTPUT BUFFER
```

```
C676  04721 072017R        STA BADDR          POINTER
C677  04722 002400          CLA              INITIALIZE
C678  04723 072020R        STA CCNT           CHARACTER COUNT
C679  04724 062045R        LDA TEMPS
C680  04725 017204R        JSB DREAD          OUTPUT
C681  04726 017132R        JSB OUTIN          SEQUENCE NUMBER
C682  04727 062271R        LDA BLANK        OUTPUT
C683  04730 017477R        JSB OUTCR          BLANK
C684  04731 036045R        ISZ TEMPS        FETCH
C685  04732 062045R        LDA TEMPS
C686  04733 017204R        JSB DREAD          STATEMENT LENGTH
C687  04734 003004          CMA,INA          SET
C688  04735 002004          INA                WORD
C689  04736 073234R        STA SLWST            COUNTER
C690  04737 036045R LIST3  ISZ TEMPS        MORE
C691  04740 037234R        ISZ SLWST          STATEMENT?
C692  04741 026752R        JMP LIST4        YES
C693  04742 066021R LIST2  LDB SBUFA        OUTPUT
C694  04743 062020R        LDA CCNT
C695  04744 117131R        JSB LISTR,I        STATEMENT
C696  04745 102501          LIA 01
C697  04746 012261R        AND HALF
C698  04747 002002          SZA                IS SW #14 SET?
C699  04750 027575R        JMP RDYPT          YES, RETURN TO READY
C700  04751 026714R        JMP LIST1
C701  04752 062045R LIST4  LDA TEMPS
C702  04753 017204R        JSB DREAD
C703  04754 012220R        AND OPMSK
C704  04755 002003          SZA,RSS          NULL OPERATOR?
C705  04756 026776R        JMP LIST5        YES
C706  04757 072050R        STA TEMP2        NO, SAVE OPERATOR
C707  04760 001727          ALF,ALF          SINGLE
C708  04761 064000          LDB 0              CHARACTER
C709  04762 042237R        ADA M21
C710  04763 002021          SSA,RSS              OPERATOR?
C711  04764 027106R        JMP LIS12        NO
C712  04765 005000          BLS              YES
C713  04766 006004          INB              LOAD
C714  04767 045100R        ADB FOPBS          SYMBOL'S
C715  04770 160001          LDA 1,I              ASCII WORD
C716  04771 001727          ALF,ALF          ADJUST
C717  04772 010074          AND MSK0         CHARACTER
C718  04773 052143R        CPA .34          " ?
C719  04774 027125R        JMP LIS14        YES
C720  04775 017477R        JSB OUTCR        NO
C721  04776 062045R LIST5  LDA TEMPS
C722  04777 017204R        JSB DREAD
C723  05000 012225R        AND OPDMK        SAVE
C724  05001 072051R        STA TEMP3          OPERAND
C725  05002 012223R        AND TYPFL        EXTRACT OPERAND TYPE
C726  05003 072362R        STA LFLAG        SET LFLAG FALSE
C727  05004 002020          SSA              FLAG BIT SET?
C728  05005 027041R        JMP LIST9        YES
C729  05006 002003          SZA,RSS          NO, NULL OPERAND?
C730  05007 026737R        JMP LIST3        YES
```

```
C731  05010 050062         CPA .7   FUNCTION
C732  05011 027034R        JMP LIST8      YES
C733  05012 040046  LIST6  ADA M5
C734  05013 002020         SSA            LETTER-DIGIT COMBINATION?
C735  05014 027017R        JMP LIST7      NO
C736  05015 003400         CCA            YES, SET
C737  05016 072362R        STA LFLAG        LFLAG FALSE
C738  05017 062051R LIST7  LDA TEMP3
C739  05020 001727         ALF,ALF        RESTORE AND
C740  05021 001722         ALF,RAL ::
C741  05022 010073         AND B177         OUTPUT
C742  05023 042160R        ADA B100
C743  05024 017477R        JSB OUTCR          LETTER
C744  05025 036362R        ISZ LFLAG      DIGIT FOLLOWS?
C745  05026 026737R        JMP LIST3      NO
C746  05027 062051R        LDA TEMP3      YES
C747  05030 010062         AND .7     RESTORE
C748  05031 042147R        ADA .43          DIGIT
C749  05032 017477R        JSB OUTCR      OUTPUT DIGIT
C750  05033 026737R        JMP LIST3
C751  05034 062162R LIST8  LDA F          OUTPUT
C752  05035 017477R        JSB OUTCR        'F'
C753  05036 062166R        LDA N          OUTPUT
C754  05037 017477R        JSB OUTCR        'N'
C755  05040 027017R        JMP LIST7
C756  05041 022263R LIST9  XOR FLGBT
C757  05042 002102         CLE,SZA        NUMBER?
C758  05043 027064R        JMP LIS10      NO
C759  05044 036045R        ISZ TEMPS      YES
C760  05045 072041R        STA SIGN       SET SIGN FLAG FALSE
C761  05046 062045R        LDA TEMPS
C762  05047 017204R        JSB DREAD
C763  05050 036045R        ISZ TEMPS
C764  05051 073130R        STA TMP2
C765  05052 062045R        LDA TEMPS
C766  05053 017204R        JSB DREAD
C001  05054 064000         LDB 0
C002  05055 063130R        LDA TMP2
C003  05056 037234R        ISZ SLWST
C004  05057 037234R        ISZ SLWST
C005  05060 002120         CLE,SSA        NEGATIVE NUMBER?
C006  05061 002300         CCE            YES, SET SIGN FLAG TRUE
C007  05062 016041R        JSB NUMOT
C008  05063 026737R        JMP LIST3
C009  05064 050056  LIS10  CPA .5         INTEGER?
C010  05065 027100R        JMP LIS11      YES
C011  05066 050062         CPA .7       NO,FUNCTION
C012  05067 002001         RSS            YES
C013  05070 027012R        JMP LIST6      NO, MUST BE A PARAMETER
C014  05071 062051R        LDA TEMP3      COMPUTE
C015  05072 001722         ALF,RAL          PRINT
C016  05073 012220R        AND UPMSK          TABLE
C017  05074 072050R        STA TEMP2            CODE
C018  05075 065122R        LDB ATAB       OUTPUT
C019  05076 017216R        JSB MCOUT        FUNCTION NAME
```

```
C020  05077 025737R        JMP LIST3
C021  05100 056045R LIS11  ISZ TEMPS      OUTPUT
C022  05101 037234R        ISZ SLWST
C023  05102 062045R        LDA TEMPS
C024  05103 017204R        JSB DREAD       INTEGER
C025  05104 017132R        JSB OUTIN
C026  05105 025737R        JMP LIST3           OPERAND
C027  05106 062271R LIS12  LDA BLANK      OUTPUT
C028  05107 017477R        JSB OUTCR         BLANK
C029  05110 066106R        LDB STTYP      OUTPUT
C030  05111 017216R        JSB MCOUT         OPERATOR
C031  05112 062212R        LDA REMOP      WAS IT
C032  05113 052050R        CPA TEMP2         A REM?
C033  05114 027173R        JMP OUTS1      YES, OUTPUT REMARK
C034  05115 062271R        LDA BLANK      NO, OUTPUT
C035  05116 025775R        JMP LIST5-1       A BLANK
C036  05117 006400  LIS13  CLB            HIGH-SPEED
C037  05120 056023R        CPB TFLAG         PUNCH?
C038  05121 027575R        JMP RDYPT         NO
C039  05122 062170R        LDA B133       YES, EMIT
C040  05123 117131R        JSB LISTR,I       TRAILER
C041  05124 027575R        JMP RDYPT
C042  05125 017477R LIS14  JSB OUTCR      OUTPUT "
C043  05126 017172R        JSB OUTST      OUTPUT QUOTE STRING
C044  05127 062143R        LDA .34        OUTPUT
C045  05130 025775R        JMP LIST5-1
C046  05131 000000  LISTR  NOP
C047*                      *
C048**  OUTPUT AN INTEGER  **
C049*                      *
C050  05132 000000  OUTIN  NOP            INTEGER IN (A)
C051  05133 064047        LDB M4         SET
C052  05134 077701R        STB DIGCT         DIGIT COUNTER
C053  05135 067251R        LDB LDVSR      SET DIVISOR
C054  05136 075400R        STB DIVSR         ADDRESS
C055  05137 006400        CLB            SET LEADING
C056  05140 076626R        STB LDZRO         ZERO FLAG
C057  05141 166400R OUTI1  LDB DIVSR,I    NEGATE
C058  05142 007004        CMB,INB           AND STORE
C059  05143 075553R        STB MIND             DIVISOR
C060  05144 007400        CCB            SET QUOTIENT
C061  05145 006004        INB               TO ZERO
C062  05146 042553R        ADA MIND       SUBTRACT DIVISOR FROM INTEGER
C063  05147 002021        SSA,RSS        NEGATIVE RESULT?
C064  05150 027145R        JMP *-3        NO, INCREMENT QUOTIENT
C065  05151 142400R        ADA DIVSR,I    YES, RECOVER REMAINDER
C066  05152 073216R        STA MCOUT         AND SAVE IT
C067  05153 060001        LDA 1
C068  05154 002002        SZA            ZERO?
C069  05155 027160R        JMP OUTI2      NO
C070  05156 052626R        CPA LDZRO      YES, LEADING ZERO?
C071  05157 027163R        JMP OUTI3      YES
C072  05160 042154R OUTI2  ADA .48        NO, COMPUTE ASCII FOR DIGIT
C073  05161 072626R        STA LDZRO      SET 'ZEROES SIGNIFICANT'
C074  05162 017477R        JSB OUTCR      OUTPUT DIGIT
```

```
C075  05163 063216R OUTI3 LDA MCOUT     RETRIEVE REMAINDER
C076  05164 035400R       ISZ DIVSR     SET FOR NEXT DIVISOR
C077  05165 037701R       ISZ DIGCT     DIVISION NECESSARY?
C078  05166 027141R       JMP OUTI1     YES
C079  05167 042154R       ADA .48       NO, COMPUTE ASCII FOR LAST
C080  05170 017477R       JSB OUTCR       DIGIT AND OUTPUT IT
C081  05171 127132R       JMP OUTIN,I
C082*                     *
C083**  OUTPUT A STRING  **
C084*                     *
C085  05172 000000  OUTST NOP             " ENTRY POINT
C086  05173 062045R OUTS1 LDA TEMPS       REM ENTRY POINT
C087  05174 017204R       JSB DREAD
C088  05175 010073       AND B177        OUTPUT SECOND CHARACTER
C089  05176 002002       SZA               OF WORD IF
C090  05177 017477R       JSB OUTCR         NOT NULL
C091  05200 036045R       ISZ TEMPS       BUMP POINTER
C092  05201 037234R       ISZ SLWST       REM COMPLETED?
C093  05202 002001       RSS             NO
C094  05203 026742R       JMP LIST2       YES
C095  05204 062045R       LDA TEMPS
C096  05205 017204R       JSB DREAD           EXTRACT
C097  05206 001727       ALF,ALF           FIRST CHARACTER
C098  05207 010073       AND B177            OF WORD
C099  05210 050054       CPA .1    EXIT
C100  05211 127172R       JMP OUTST,I     IF A
C101  05212 052245R       CPA STCON       CLOSING
C102  05213 127172R       JMP OUTST,I           QUOTE
C103  05214 017477R       JSB OUTCR       OUTPUT
C104  05215 027173R       JMP OUTS1         CHARACTER
C105*                                   *
C106**  LIST A MULTICHARACTER SYMBOL   **
C107*                                   *
C108  05216 000000  MCOUT NOP
C109  05217 160001  MCOU1 LDA 1,I         LOAD INFORMATION WORD
C110  05220 012220R       AND OPMSK       COMPARE WITH
C111  05221 052050R       CPA TEMP2         OPERATOR CODE
C112  05222 027231R       JMP MCOU2       EQUAL
C113  05223 160001       LDA 1,I         UNEQUAL,
C114  05224 010070       AND .15            COMPUTE
C115  05225 040056       ADA .5               ENTRY
C116  05226 001100       ARS                    LENGTH
C117  05227 044000       ADB 0           COMPUTE ADDRESS OF NEXT ENTRY
C118  05230 027217R       JMP MCOU1
C119  05231 160001  MCOU2 LDA 1,I         COMPUTE
C120  05232 010070       AND .15            ENTRY
C121  05233 003004       CMA,INA              LENGTH
C122  05234 037701R       STA DIGCT              AND SAVE IT
C123  05235 006104       CLE,INB         SET FOR FIRST CHARACTER
C124  05236 076051R       STB TEMP3       SAVE SYMBOL ADDRESS
C125  05237 162051R MCOU3 LDA TEMP3,I    LOAD WORD
C126  05240 002041       SEZ,RSS         FIRST CHARACTER?
C127  05241 001727       ALF,ALF         YES, POSITION IT
C128  05242 010073       AND B177        EXTRACT CHARACTER
C129  05243 017477R       JSB OUTCR       OUTPUT IT
```

```
C130   05244 002240          SEZ,CME         SET FOR NEXT CHARACTER
C131   05245 036051R         ISZ TEMP3       MOVE TO NEXT WORD OF SYMBOL
C132   05246 037701R         ISZ DIGCT       MORE CHARACTERS?
C133   05247 027257R         JMP MCOU3       YES
C134   05250 127216R         JMP MCOUT,I
C135*
C136*
C137   05251 005252R LDVSR   DEF *+1
C138   05252 023420          DEC 10000
C139   05253 001750          DEC 1000
C140   05254 000144          DEC 100
C141   05255 000012          DEC 10
C142*
C143*
C144   03534           SFLAG EQU ARRYS
C145   04362           TABLE EQU PRGIN
C146   00055           LNGTH EQU TEMPS+8
C147   04400           SMEND EQU INTCK
C148   04626           SLENG EQU OVCHK
C149   04553           TBLPT EQU FNDPS
C150   04603           TSPTR EQU CLPRG
C151   04626           INTGR EQU OVCHK
C152   04362           LFLAG EQU PRGIN
C153   04400           DIVSR EQU INTCK
C154   04626           LDZRO EQU OVCHK
C155   04553           MIND  EQU FNDPS
```

```
C157*              **********************
C158*              PHASE 2 OF THE COMPILER
C159*              **********************
C160*
C161*              THIS PHASE HAS THE FOLLOWING 3 FUNCTIONS:
C162*              1. SYMBOL TABLE CONSTRUCTION
C163*              2. FOR LOOP CHECKING
C164*              3. ARRAY STORAGE ALLOCATION
C165*
C166  05256 062010C MFASE LDA PBPTR     NULL
C167  05257 072007R       STA FDISK     ADDR. OF START OF ARRAY STORAGE
C168  05260 052007C       CPA PBUFF       PROGRAM?
C169  05261 027575R       JMP RDYPT       YES
C170  05262 062013R       LDA ASBTB
C171  05263 072006R       STA FCORE      NO, SET FOR-TABLE POINTER
C172  05264 062002R       LDA FWAM
C173  05265 072056R       STA COML      INITIALIZE COMMON POINTER
C174  05266 062011R       LDA SYMTA
C175  05267 072010R       STA SYMTF     INITIALIZE SYMBOL TABLE POINTER
C176  05270 062007C       LDA PBUFF
C177  05271 072022R       STA MPTR      INITIALIZE PROGRAM POINTER
C178  05272 062022R MLOP1 LDA MPTR
C179  05273 017204R       JSB DREAD
C180  05274 064000        LDB A
C181  05275 076033R       STB .LNUM     SET LINE NUMBER
C182  05276 066022R       LDB MPTR
C183  05277 036022R       ISZ MPTR
C184  05300 062022R       LDA MPTR
C185  05301 017204R       JSB DREAD
C186  05302 044000        ADB A         COMPUTE LOCATION OF NEXT
C187  05303 077266R       STB MNPTR      STATEMENT AND STORE THIS
C188  05304 036022R       ISZ MPTR
C189  05305 062022R       LDA MPTR
C190  05306 017204R       JSB DREAD    FETCH THE FIRST WORD IN THE
C191  05307 001727  MLO10 ALF,ALF       STATEMENT AND ITS TYPE
C192  05310 010073        AND B177   :::
C193  05311 072034R       STA TYPE
C194  05312 052152R       CPA .46        MAT STATEMENT?
C195  05313 027325R       JMP MLO12      YES
C196  05314 052140R       CPA .30        NO, REM STATEMENT?
C197  05315 076022R       STB MPTR   YES,SET TO SKIP IT
C198  05316 052156R       CPA .50    WRITE STATEMENT?
C199  05317 027321R       JMP *+2     YES,SET TO SKIP IT
C200  05320 052147R       CPA .45        NO, PRINT STATEMENT?
C201  05321 076022R       STB MPTR       YES, SET TO SKIP IT
C202  05322 003400        CCA            NO, SET
C203  05323 072057R       STA MWDNO        'FIRST VARIABLE'
C204  05324 027541R       JMP MLOP2+1        FLAG
C205*
C206  05325 062022R MLO12 LDA MPTR       SEEK
C207  05326 002004        INA              SUBSIDIARY
C208  05327 017204R       JSB DREAD              STATEMENT
C209  05330 027307R       JMP MLO10            TYPE
C210*
C211  05331 012176R MLO13 AND MSK1       YES, ISOLATE OPERAND
```

```
C212  05332 066022R        LDB MPTR      INDEX THE PROGRAM POINTER BY
C213  05333 002003         SZA,RSS       AN AMOUNT APPROPRIATE TO THE
C214  05334 044055         ADB .2        OPERAND. THE FOLLOWING APPLIES
C215  05335 050056         CPA .3        OPERAND = 0  ADD 2 TO POINTER
C216  05336 006004         INB           OPERAND =3  ADD 1 TO POINTER
C217  05337 076022R        STB MPTR
C218*
C219  05340 036022R MLOP2  ISZ MPTR       INCREMENT WORD-OF-STATEMENT PTR
C220  05341 062022R        LDA MPTR      STATEMENT
C221  05342 053266R        CPA MNPTR       EXHAUSTED?
C222  05343 027433R        JMP MLOP5     YES
C223  05344 062022R        LDA MPTR
C224  05345 017204R        JSB DREAD   GET STAEMENT TYPE
C225  05346 012220R        AND OPMSK
C226  05347 052215R        CPA PRNOP   PRINT?
C227  05350 027431R        JMP MLO14   YES, SKIP TO NEXT STATMENT
C228  05351 052214R        CPA WROP    WRITE?
C229  05352 027431R        JMP MLO14   YES, SKIP TO NEXT STATMENT
C230  05353 062022R        LDA MPTR     NO
C231  05354 017204R        JSB DREAD
C232  05355 002020         SSA           'CONSTANT' OPERAND?
C233  05356 027331R        JMP MLO13     YES
C234  05357 012176R        AND MSK1      NO
C235  05360 002003         SZA,RSS       NULL OPERAND?
C236  05361 027340R        JMP MLOP2     YES
C237  05362 072045R        STA MBUX1     NO, SAVE IT
C238  05363 010062         AND .7         PROGRAMMER-DEFINED
C239  05364 050062         CPA .7           FUNCTION?
C240  05365 027442R        JMP MLOP6     YES
C241  05366 040047         ADA M4        NO
C242  05367 002020         SSA           ARRAY VARIABLE?
C243  05370 027463R        JMP MLOP7     YES
C244  05371 062045R        LDA MBUX1     NO, SIMPLE VARIABLE
C245  05372 017613R        JSB SSYMT       ALREADY IN
C246  05373 006021         SSB,RSS         SYMBOL TABLE?
C247  05374 027403R        JMP MLOP3     YES
C248  05375 062263R        LDA MNEG      NO
C249  05376 066264R        LDB MNEG+1    ENTER
C250  05377 072046R        STA MBUX1+1     IT WITH
C251  05400 076047R        STB MBUX1+2       'UNDEFINED'
C252  05401 060050         LDA M3              VALUE
C253  05402 017660R        JSB ESYMT
C254  05403 066034R MLOP3  LDB TYPE
C255  05404 062045R        LDA MBUX1
C256  05405 056143R        CPB .34       NEXT STATEMENT?
C257  05406 027420R        JMP MLOP4     YES
C258  05407 056142R        CPB .33       NO, FOR STATEMENT?
C259  05410 036057R        ISZ MWDNO     YES, FIRST VARIABLE?
C260  05411 027340R        JMP MLOP2     NO
C261  05412 036006R        ISZ FCORE     DEMAND
C262  05413 066006R        LDB FCORE       SPACE
C263  05414 056010R        CPB SYMTF         FOR NEW
C264  05415 027670R        JMP MER8-1          ENTRY
C265  05416 172006R        STA FCORE,I   SAVE VARIABLE NAME
C266  05417 027340R        JMP MLOP2
```

```
C267*
C268  05420 066006R MLOP4 LDB FCORE      FOR-TABLE
C269  05421 056013R       CPB ASBTB       EMPTY?
C270  05422 015274R       JSB ERROR      YES
C271  05423 152006R MER3  CPA FCORE,I    NO, MATCH LATEST ENTRY?
C272  05424 002001       RSS            YES
C273  05425 027422R       JMP MER3-1     NO
C274  05426 044052       ADB M1         REMOVE
C275  05427 076006R       STB FCORE       MATCHED
C276  05430 027340R       JMP MLOP2         ENTRY
C277*
C278  05431 063266R MLO14 LDA MNPTR
C279  05432 072022R       STA MPTR
C280  05433 052010C MLOP5 CPA PBPTR      PROGRAM EXHAUSTED?
C281  05434 002001       RSS            YES
C282  05435 027272R       JMP MLOP1      NO
C283  05436 062054R       LDA TYPE       YES
C284  05437 052144R       CPA .37        END STATEMENT?
C285  05440 027561R       JMP M1LOP      YES
C286  05441 015274R       JSB ERROR      NO
C287  05442 062022R MLOP6 LDA MPTR       ISOLATE
C288  05443 017204R       JSB DREAD
C289  05444 013220R       AND OPMSK        PRECEDING OPERATOR
C290  05445 052211R       CPA DEFOP      'DEF' ?
C291  05446 002001       RSS            YES
C292  05447 027340R       JMP MLOP2     NO GO TO PROCESS NEXT WORD
C293  05450 062045R       LDA MBUX1     SEARCH SYMBOL TABLE FOR
C294  05451 017613R       JSB SSYMT        THE FUNCTION
C295  05452 006021       SSB,RSS
C296  05453 015274R       JSB ERROR     FOUND. ERROR MULTIPLY DEFINED
C297  05454 062022R MER4  LDA MPTR
C298  05455 040056       ADA .3        ENTER THE FUNCTION INTO THE
C299  05456 052263R       IOR FLGBT     ADD V.A. FLAG
C300  05457 072046R       STA MBUX1+1   SYMBOL TABLE TOGETHER WITH
C301  05460 060051       LDA M2        ITS ENTRY POINT IN THE SOURCE
C302  05461 017660R       JSB ESYMT     CODE
C303  05462 027340R       JMP MLOP2     GO TO PROCESS THE NEXT WORD
C304*
C305  05463 070001 MLOP7 STA 1
C306  05464 062054R       LDA TYPE
C307  05465 052136R       CPA .27        DIM STATEMENT?
C308  05466 027500R       JMP MLOP8      YES
C309  05467 052137R       CPA .28        NO, COM STATEMENT?
C310  05470 027500R       JMP MLOP8      YES
C311  05471 017701R       JSB MSYMT      NO, LOOK FOR IT IN SYMBOL TABLE
C312  05472 027340R       JMP MLOP2      FOUND
C313  05473 002400       CLA            NOT THERE
C314  05474 072046R       STA MBUX1+1    ENTER IT WITH
C315  05475 072047R       STA MBUX1+2      DIMENSIONS AND
C316  05476 072050R       STA MBUX1+3        DIMENSIONALITY
C317  05477 027542R       JMP MLOP0            UNDEFINED
C318*
C319  05500 056022R MLOP8 ISZ MPTR      PROCESS COM OR DIM STMT
C320  05501 056022R       ISZ MPTR
C321  05502 062022R       LDA MPTR     PICK UP FIRST DIMENSION
```

```
C322  05503 017204R         JSB DREAD
C323  05504 001727          ALF,ALF       SHIFT TO M. S. PART OF WORD
C324  05505 054050          CPB M3        IS THIS A SINGLE DIMENSION ARRAY
C325  05506 027516R         JMP *+8       YES, JUMP
C326  05507 056022R         ISZ MPTR   . NO, INDEX POINTER TO THE LOC.
C327  05510 056022R         ISZ MPTR      OF SECOND DIMENSION AND PACK
C328  05511 073150R         STA TMP2    INTO A WITH
C329  05512 062022R         LDA MPTR
C330  05513 017204R         JSB DREAD
C331  05514 033150R         IOR TMP2    THE FIRST DIMENSION
C332  05515 002001          RSS
C333  05516 050054          IOR .1
C334  05517 072047R         STA MBOX1+2   SET UP TO STORE PACKED
C335  05520 072050R         STA MBOX1+3   DIMENSIONS IN FORMAL AND ACTUAL
C336  05521 002400          CLA            SLOTS AND UNDEFINED FLAG IN
C337  05522 072046R         STA MBOX1+1   STORAGE ALLOCATION SLOT
C338  05523 017701R         JSB MSYMT      IN SYMBOL TABLE?
C339  05524 027545R         JMP MLOP9      NO
C340  05525 062034R         LDA TYPE       YES
C341  05526 052137R         CPA .26
C342  05527 002001          RSS            IS STMT A COM
C343  05530 027542R         JMP MLOP0      NO, JUMP
C344  05531 062047R         LDA MBOX1+2   YES PICK UP PACKED DIMENSIONS
C345  05532 017130R         JSB MD1M      COMPUTE STORAGE REQUIRED
C346  05533 064000          LDB A
C347  05534 062056R         LDA COML       POINTER TO NEXT FREE LOC IN COM
C348  05535 052263R         IOR FLGBT      ADD V.A. FLAG
C349  05536 072046R         STA MBOX1+1    STORE IN STORAGE ALLOCATION SLOT
C350  05537 012222R         AND INF
C351  05540 040001          ADA B
C352  05541 072056R         STA COML      UPDATE PTR BY AMT OF STORAGE REQ.
C353  05542 060047  MLOP0   LDA M4        ENTER THE FOUR WORD ENTRY
C354  05543 017660R         JSB ESYMT     PREVIOUSLY SET UP IN MBOX1 INTO
C355  05544 027340R         JMP MLOP2     SYMBOL TABLE AND CONTINUE
C356*
C357  05545 044055  MLOP9   ADB .2          CHECK THE FORMAL DIMENSIONS
C358  05546 160001          LDA 1,I       LOCATION TO SEE IF THE DIMENSION
C359  05547 002002          SZA            IS ALREADY DEFINED
C360  05550 015274R         JSB ERROR     ERROR, DOUBLY DIMENSIONED
C361  05551 062034R MER5    LDA TYPE
C362  05552 052137R         CPA .26        COM STMT?
C363  05553 026331R         JMP SYNE3-1      ERROR MISPLACED COM. STMT.
C364  05554 062047R         LDA MBOX1+2
C365  05555 170001          STA 1,I       STORE THESE DIMENSIONS IN FORMAL
C366  05556 005004          INB           AND ACTUAL SLOTS IN SYMBOL TABLE
C367  05557 170001          STA 1,I       ENTRY
C368  05560 027340R         JMP MLOP2     GO TO PROCESS NEXT WORD
C369*
C370*                  THE SECTION WHICH FOLLOWS CHECKS
C371*                  THAT ALL FOR LOOPS HAVE BEEN
C372*                  TERMINATED, ASSIGNS THE STANDARD
C373*                  DIMENSIONS TO UNDIMENSIONED ARRAYS
C374*                  AND MAKES STORAGE ASSIGNMENTS FOR
C375*                  ALL ARRAYS WHICH DO NOT APPEAR IN
C376*                  A COM STMT
```

```
C377*
C378  05561 062006R M1LOP LDA FCORE      ALL FORS
C379  05562 052013R      CPA ASBTB          MATCHED?
C380  05563 002001       RSS              YES
C381  05564 015274R      JSB ERROR        NO
C382  05565 066010R MER6  LDB 'SYMTF
C383*
C384  05566 056011R M2LOP CPB SYMTA       MORE SYMBOLS?
C385  05567 027642R      JMP M4LOP        NO
C386  05570 160001       LDA 1,I          YES
C387  05571 010062       AND .7   ::      ACCOUNT FOR
C388  05572 044055       ADB .2             A FUNCTION
C389  05573 050062       CPA .7   ::    IS IT?
C390  05574 027566R      JMP M2LOP        YES
C391  05575 006004       INB              NO, ACCOUNT FOR
C392  05576 040047       ADA M4             SIMPLE VARIABLE
C393  05577 002025       SSA,INA,RSS      IS IT?
C394  05600 027566R      JMP M2LOP        YES
C395  05601 002003       SZA,RSS          NO, # OF SUBSCRIPTS KNOWN?
C396  05602 015274R      JSB ERROR        NO
C397  05603 002004  MER10 INA              SAVE
C398  05604 072046R      STA MBUX1+1        FLAG
C399  05605 076045R      STB MBUX1        SAVE POINTER
C400  05606 160001       LDA 1,I          DEFINED
C401  05607 002002       SZA                ARRAY?
C402  05610 027617R      JMP M3LOP        YES
C403  05611 063657R      LDA STDIM        NO, LOAD
C404  05612 036046R      ISZ MBUX1+1        APPROPRIATE
C405  05613 040064       ADA .9               STANDARD DIMENSIONS
C406  05614 170001       STA 1,I          RECORD AS
C407  05615 044052       ADB M1             FORMAL AND ACTUAL
C408  05616 170001       STA 1,I              DIMENSIONS
C409  05617 017130R M3LOP JSB MDIM        SAVE STORAGE
C410  05620 072046R      STA MBUX1+1        REQUIREMENT
C411  05621 066045R      LDB MBUX1        LOAD
C412  05622 044051       ADB M2             ADDRESS OF
C413  05623 160001       LDA 1,I              ELEMENT SPACE
C414  05624 012222R      AND INF      REMOVE V.A. BIT
C415  05625 050055       CPA .2       DEFINED IN COM?
C416  05626 027640R      JMP MER7       YES
C417  05627 052007R      LDA FDISK       NO, USE CURRENT
C418  05630 032263R      IOR FLGBT     ADD V.A. FLAG
C419  05631 170001       STA 1,I          FREE-CORE ADDRESS
C420  05632 012222R      AND INF
C421  05633 103101       CLO
C422  05634 042046R      ADA MBUX1+1    UPDATE FREE-CORE
C423  05635 072007R      STA FDISK              ADDRESS
C424  05636 102201       SOC              OUT OF SPACE ?
C425  05637 015274R      JSB ERROR        YES
C426  05640 044056  MER7  ADB .3          NO, ADVANCE POINTER
C427  05641 027566R      JMP M2LOP          TO NEXT ENTRY
C428*
C429  05642 066010C M4LOP LDB PBPTR       INITIALIZE ALL
C430  05643 060001       LDA 1
C431  05644 003000       CMA
```

```
C432  05645 042007R        ADA FDISK      ARRAY ELEMENTS
C433  05646 002020         SSA
C434  05647 026173R        JMP XEC         TO 'UNDEFINED'
C435  05650 062263R        LDA MNEG
C436  05651 017055R        JSB STBI
C437  05652 006004         INB
C438  05653 062264R        LDA MNEG+1
C439  05654 017055R        JSB STBI
C440  05655 006004         INB
C441  05656 027643R        JMP M4LOP+1
C442*
C443  05657 005001  STDIM GCT 5001
```

```
C445*                ******************************
C446*                ENTER SYMBOL TABLE SUBROUTINE
C447*                ******************************
C448*
C449*                TRANSFER -(A) WORDS FROM THE BUFFER ADDRESSED
C450*                BY MBUF TO THE TOP OF THE SYMBOL TABLE.
C451*
C452  05660 000000  ESYMT NOP
C453  05661 073254R        STA MBIN1     SAVE NEGATIVE OF LENGTH OF ENTRY
C454  05662 042010R        ADA SYMTF
C455  05663 072010R        STA SYMTF     MOVE SYMBOL TABLE START LOCATOR
C456  05664 073321R        STA MBIN2     UP BY THE LENGTH OF ENTRY
C457  05665 003004        CMA,INA       CHECK THAT THE SYMBOL TABLE AND
C458  05666 042006R        ADA FCORE      FOR TABLE DO NOT OVERLAP
C459  05667 002021        SSA,RSS
C460  05670 016274R        JSB ERROR     OVERLAP ERROR
C461  05671 067722R MER8   LDB MBUF      POINTER TO REQD ENTRY
C462  05672 160001        LDA 1,I       TRANSFER ENTRY TO THE SYMBOL
C463  05673 173521R        STA MBIN2,I   TABLE
C464  05674 006004        INB
C465  05675 037321R        ISZ MBIN2
C466  05676 037254R        ISZ MBIN1
C467  05677 027672R        JMP MER8+1
C468  05700 127660R        JMP ESYMT,I   RETURN
C469*
C470*           ***********************************************
C471*           SUBROUTINE TO SEARCH SYMBOL TABLE FOR AN ARRAY
C472*           ***********************************************
C473  05701 000000  MSYMT NOP             B GIVES ARRAY TYPE -3 = 1 DIM,
C474  05702 077254R        STB MBIN1     -2 = 2DIM, -1 = UNDIMENSIONED
C475  05703 062045R        LDA MBUX1     LOAD IDENTIFIER
C476  05704 017613R        JSB SSYMT        SEARCH THE SYMBOL TABLE
C477  05705 006021        SSB,RSS
C478  05706 127701R        JMP MSYMT,I   FOUND, RETURN
C479  05707 037254R        ISZ MBIN1     IF ARRAY UNDIMENSIONED
C480  05710 002001        RSS
C481  05711 027720R        JMP MSYM      JUMP TO NOT FOUND EXIT
C482  05712 037254R        ISZ MBIN1     SET UP TO CHECK THAT ARRAY DOES
C483  05713 040055        ADA .2        NOT APPEAR IN THE TABLE WITH
C484  05714 040052        ADA M1        DIFFERENT DIMENSIONS. CHANGE
C485  05715 017613R        JSB SSYMT        TYPE 2 TO 1 & TYPE 1 TO 2 AND
C486  05716 006021        SSB,RSS       SEARCH AGAIN
C487  05717 016274R        JSB ERROR     FOUND, INCONSISTENT DIMENSIONS
C488  05720 037701R MSYM   ISZ MSYMT     NOT FOUND, INCREMENT RETURN
C489  05721 127701R        JMP MSYMT,I   ADDRESS AND RETURN
C490*
C491*
C492  05722 000045R MBUF   DEF TEMPS
C493  00045         MBUX1 EQU TEMPS
C494  01234         MBIN1 EQU SLWST
C495  01321         MBIN2 EQU RSCHK
C496  00022         MPTR  EQU SBPTR
C497  01266         MNPTR EQU UPCHK
C498  00056         COML  EQU TEMPS+9
C499  00057         MWDNO EQU TEMPS+10
```

```
C500  05701          DIGCI EQU MSYMT
```

```
C502*
C503***                          ***
C504**  EVALUATE A FORMULA       **
C505***                          ***
C506*
C507  05725 000000  FORMX NOP            FORMULA BEGINS IN (TEMPS)
C508  05724 006400        CLB            INITIALIZE OPERATOR
C509  05725 017234R       JSB SLWST        STACK
C510  05726 017062R FORM1 JSB LDTMP      FETCH OPERAND
C511  05727 036045R       ISZ TEMPS      SET FOR NEXT WORD OF FORMULA
C512  05730 012225R       AND OPDMK      EXTRACT OPERAND
C513  05731 072053R       STA TEMPS+6      AND SAVE IT
C514  05732 002005        SZA,RSS        NULL OPERAND?
C515  05733 027746R       JMP FORM2      YES
C516  05734 017243R       JSB BHSTP      SET STACK FOR OPERAND ADDRESS
C517  05735 002020        SSA            FLAG BIT SET?
C518  05736 026015R       JMP FORM4      YES
C519  05737 017613R       JSB SSYMT        FETCH OPERAND ADDRESS
C520  05740 006007        INB,SZB,RSS    EXISTANT?
C521  05741 027302R       JMP E8-1         NO
C522  05742 010062        AND .7     YES ::
C523  05743 050062        CPA .7       FUNCTION?
C524  05744 026033R       JMP FORM6      YES
C525  05745 176030R       STB HSTPT,I    NO, STACK OPERAND ADDRESS
C526  05746 017062R FORM2 JSB LDTMP      FETCH
C527  05747 012220R       AND OPMSK        OPERATOR
C528  05750 001727        ALF,ALF        POSITION IT
C529  05751 001200        RAL  ::
C530  05752 064000        LDB 0          LOAD ADDRESS OF
C531  05753 046100R       ADB FOPBS        OPERATOR'S INFORMATION WORD
C532  05754 040043        ADA M8         NON-FORMULA
C533  05755 002020        SSA              OPERATOR?
C534  05756 006400        CLB            YES
C535  05757 042241R       ADA D55        NO, NON-FORMULA
C536  05760 002021        SSA,RSS          OPERATOR?
C537  05761 006400        CLB            YES
C538  05762 002400        CLA            NO
C539  05763 160001        LDA 1,I        LOAD INFORMATION WORD
C540  05764 012176R       AND MSK1       SAVE
C541  05765 072054R       STA TEMPS+7      PRECEDENCE
C542  05766 120001        XOR 1,I        SAVE
C543  05767 072053R       STA TEMPS+6      IDENTIFICATION
C544  05770 027776R       JMP FOR11
C545  05771 172026R FORM0 STA TSTPT,I    STACK HIGH WORD
C546  05772 062026R       LDA TSTPT      STACK OPERAND
C547  05773 172030R       STA HSTPT,I      ADDRESS
C548  05774 002004        INA            STORE
C549  05775 174000        STB 0,I          LOW WORD
C550  05776 162027R FOR11 LDA LSTPT,I    DOES OPERATOR
C551  05777 010074        AND MSK0         ON TOP OF
C552  06000 003000        CMA                OPERATOR STACK
C553  06001 042054R       ADA TEMPS+7          HAVE HIGHER
C554  06002 002020        SSA                    PRECEDENCE?
C555  06003 025136R       JMP FORM9      YES, EXECUTE IT
C556  06004 002001        RSS            NO
```

```
C557  06005 036027R FOR10 ISZ LSTPT
C558  06006 066054R       LDB TEMPS+7    RETRIEVE PRECEDENCE
C559  06007 046235R       ADB M15        NO, LEFT PARENTHESIS
C560  06010 006020        SSB              OR LEFT BRACKET?
C561  06011 044070        ADB .15    NO,RESTORE PRECEDENCE
C562  06012 046053R       ADB TEMPS+6    COMBINE IDENTIFICATION
C563  06013 017234R       JSB SLWST        WITH PRECEDENCE AND STACK
C564  06014 027726R       JMP FORM1
C565  06015 052263R FORM4 CPA FLGBT      CONSTANT?
C566  06016 026024R       JMP FORM5      YES
C567  06017 010062        AND .7     NO,PREDIFINED
C568  06020 050062        CPA .7     FUNCTION
C569  06021 026113R       JMP FORM7      YES
C570  06022 066056R       LDB TEMPS+9    NO, MUST BE A
C571  06023 027745R       JMP FORM2-1      PARAMETER
C572  06024 066045R FORM5 LDB TEMPS      LOAD CONSTANT ADDRESS
C573  06025 036045R       ISZ TEMPS      MOVE POINTER TO
C574  06026 036045R       ISZ TEMPS        NEXT CODE WORD
C575  06027 060001        LDA 1
C576  06030 032263R       IOR FLGBT    ADD V.A. FLAG
C577  06031 064000        LDB A
C578  06032 027745R       JMP FORM2-1
C579  06033 076053R FORM6 STB TEMPS+6    SAVE SYMBOL TABLE POINTER
C580  06034 066026R       LDB TSTPT      SAVE CURRENT POINTER
C581  06035 017234R       JSB SLWST        TO TEMPORARY STACK
C582  06036 166053R       LDB TEMPS+6,I
C583  06037 017234R       JSB SLWST      SAVE FUNCTION ADDRESS
C584  06040 063726R       LDA FORMX      SAVE CURRENT
C585  06041 172030R       STA HSTPT,I      FORMX RETURN ADDRESS
C586  06042 017723R       JSB FORMX      EVALUATE THE PARAMETER
C587  06043 036045R       ISZ TEMPS      UPDATE FORMULA POINTER
C588  06044 036045R       ISZ TEMPS        PAST RIGHT PARENTHESIS
C589  06045 066045R       LDB TEMPS      SWITCH
C590  06046 162027R       LDA LSTPT,I      FORMULA POINTER
C591  06047 012222R       AND INF
C592  06050 072045R       STA TEMPS          TO FUNCTION'S
C593  06051 176027R       STB LSTPT,I          FORMULA
C594  06052 066056R       LDB TEMPS+9    SET
C595  06053 162030R       LDA HSTPT,I      PARAMETER POINTER
C596  06054 036027R       ISZ LSTPT          TO NEW PARAMETER,
C597  06055 036030R       ISZ HSTPT            SAVING PREVIOUS
C598  06056 176027R       STB LSTPT,I            SETTING ON
C599  06057 072056R       STA TEMPS+9              LOW-CORE STACK
C600  06060 052026R       CPA TSTPT      PROTECT PARAMETER IF
C601  06061 017321R       JSB RSCHK        ON TEMPORARY STACK
C602  06062 017723R       JSB FORMX      EVALUATE FUNCTION
C603  06063 162027R       LDA LSTPT,I    RESTORE OLD
C604  06064 072056R       STA TEMPS+9      PARAMETER POINTER
C605  06065 066027R       LDB LSTPT      CUT BACK
C606  06066 044050        ADB M3           LOW-CORE
C607  06067 076027R       STB LSTPT          STACK
C608  06070 006004        INB            RESTORE ORIGINBL
C609  06071 160001        LDA 1,I          TEMPORARY STACK
C610  06072 072026R       STA TSTPT          POINTER
C611  06073 006004        INB            RESTORE
```

```
C612  06074 160001        LDA 1,I         ORIGINAL
C613  06075 012222R       AND INF
C614  06076 072045R       STA TEMPS         FORMULA POINTER
C615  06077 101100        SWP
C616  06100 017252R       JSB STTOP      POP RESULT
```

```
C618*
C619**  PRE-DEFINED FUNCTIONS RETURN HERE WITH RESULT
C620*
C621  06101 172026R FUR12 STA TSTPT,I   STORE HIGH WORD
C622  06102 052026R       LDA TSTPT
C623  06103 002004        INA            STORE
C624  06104 174000        STB 0,I          LOW WORD
C625  06105 056030R       ISZ HSTPT
C626  06106 156030R       LDB HSTPT,I    RESTORE FORMX
C627  06107 077723R       STB FORMX        RETURN ADDRESS
C628  06110 040052        ADA M1         STACK ADDRESS
C629  06111 172030R       STA HSTPT,I      OF RESULT
C630  06112 027746R       JMP FORM2
C631  06113 062053R FORM7 LDA TEMPS+6    COMPUTE
C632  06114 001727        ALF,ALF
C633  06115 001722        ALF,RAL      FUNCTION
C634  06116 010071        AND .31
C635  06117 042104R       ADA PDFBS          ADDRESS
C636  06120 164000        LDB 0,I
C637  06121 017234R       JSB SLWST      SAVE FUNCTION ADDRESS
C638  06122 063723R       LDA FORMX      SAVE CURRENT
C639  06123 172030R       STA HSTPT,I      FORMX RETURN ADDRESS
C640  06124 017723R       JSB FORMX      EVALUATE THE PARAMETER
C641  06125 036045R       ISZ TEMPS      UPDATE FORMULA POINTER
C642  06126 036045R       ISZ TEMPS        PAST RIGHT PARENTHESIS
C643  06127 166027R       LDB LSTPT,I    POP
C644  06130 003400        CCA              FUNCTION
C645  06131 042027R       ADA LSTPT          ENTRY
C646  06132 072027R       STA LSTPT            ADDRESS
C647  06133 077660R       STB ESYMT      SAVE
C648  06134 017252R       JSB STTOP      POP PARAMETER
C649  06135 127660R       JMP ESYMT,I    EVALUATE FUNCTION
C650  06136 162027R FORM9 LDA LSTPT,I    UNSTACK
C651  06137 007400        CCB              OPERATOR
C652  06140 046027R       ADB LSTPT          INFORMATION
C653  06141 076027R       STB LSTPT            WORD
C654  06142 001727        ALF,ALF        COMPUTE
C655  06143 010073        AND B177         SUBROUTINE
C656  06144 042103R       ADA ARBAS          ADDRESS
C657  06145 124000        JMP 0,I        EXECUTE
C658**
C659***  EXECUTION BRANCH TABLE  **
C660**
C661  06146 006406R XECTB DEF ELET       LET
C662  06147 006233R       DEF XEC4       DIM
C663  06150 006233R       DEF XEC4       COM
C664  06151 006233R       DEF XEC4       DEF
C665  06152 006233R       DEF XEC4       REM
C666  06153 006410R       DEF EGOTO      GO TO
C667  06154 006412R       DEF EIF        IF
C668  06155 006422R       DEF EFOR       FOR
C669  06156 006515R       DEF ENEXT      NEXT
C670  06157 006560R       DEF EGOSB      GOSUB
C671  06160 006571R       DEF ERTRN      RETURN
C672  06161 001575R       DEF RDYPT      END
```

```
C673  06162 001575R        DEF RDYPT      STOP
C674  06163 005600R        DEF EWAIT      WAIT
C675  06164 005617R        DEF ECALL      CALL
C676  06165 006233R        DEF XEC4       DATA
C677  06166 005647R        DEF EREAD      READ
C678  06167 005707R        DEF EPRIN      PRINT
C001  06170 007074R        DEF EINPT      INPUT
C002  06171 007112R        DEF ERSTR      RESTORE
C003  06172 011465R        DEF EMAT       MAT
C004*    *   *    *    *    *    *    *              ****
C005*
C006* ************************
C007****                     ***
C008***  EXECUTE THE PROGRAM  ***
C009****                     ***
C010* ************************
C011*
C012**
C013***INITIALIZE FOR OUTPUT  **
C014**
C015  06173 002400  XEC   CLA            SET COUNTER FOR
C016  06174 072034R       STA TYPE         CHARACTERS OUTPUTTED
C017  06175 072043R       STA XH         INITIALIZE
C018  06176 002004        INA              RANDOM
C019  06177 072044R       STA XL             VARIABLE
C020  06200 060052        LDA M1       SET EXECUTION
C021  06201 073264R       STA XIT        EXIT FLAG
C022**
C023***  INITIALIZE THE DATA POINTER  **
C024**
C025  06202 003400        CCA            SET
C026  06203 072037R       STA DCCNT        'NO
C027  06204 072035R       STA DSTRT          DATA'
C028  06205 066007C       LDB PBUFF            CONDITION
C029  06206 076036R       STB NXTDT
C030  06207 162115R       LDA ADATA,I    SEARCH FOR FIRST
C031  06210 016502R       JSB STSRH        DATA STATEMENT
C032  06211 026214R       JMP XEC2       NONE FOUND
C033  06212 076035R       STB DSTRT      SAVE STATEMENT LOCATION
C034  06213 016271R       JSB SETDP      SET DATA POINTER
C035**
C036***  INITIALIZE STACK POINTERS  **
C037**
C038  06214 066010R XEC2  LDB SYMTF      INITIALIZE
C039  06215 076030R       STB HSTPT        POINTERS TO
C040  06216 066006R       LDB FCORE          'HIGH CORE' STACK,
C041  06217 076026R       STB TSTPT            'TEMPORARY'
C042  06220 046134R       ADB .25                STACK, AND
C043  06221 076012R       STB LSTAK                'LOW CORE'
C044  06222 076027R       STB LSTPT                  STACK
C045  06223 007000        CMB            DO
C046  06224 046030R       ADB HSTPT        STACKS
C047  06225 005020        SSB                MEET?
C048  06226 027240R       JMP E1         YES
C049  06227 066226R       LDB RMODE      NO, SHIFT TO
```

EXECUTE THE PROGRAM

```
C050  06230 077131R       STB LISTR       RUN MODE
C051  06231 066007C       LDB PBUFF     BEGIN
C052  06232 026246R       JMP XEC5        EXECUTION
```

```
C054**
C055***  FIND NEXT STATEMENT TO BE EXECUTED  **
C056**
C057  06233 062032R XEC4  LDA NXTST     NEXT STATEMENT NUMBER
C058  06234 066031R       LDB PRADD     PROSPECTIVE ADDRESS
C059  06235 072032R       STA NXTST
C060  06236 017051R       JSB LDBI
C061  06237 052032R       CPA NXTST      DESIRED STATEMENT?
C062  06240 026247R       JMP XEC6      YES
C063  06241 066007C       LDB PBUFF     NO, FIND
C064  06242 062032R       LDA NXTST
C065  06243 015553R       JSB FNDPS       STATEMENT
C066  06244 000000        NOP           NON-EXISTENT
C067  06245 015274R       JSB ERROR       STATEMENT
C068  06246 017051R XEC5  JSB LDBI       SAVE NEW
C069  06247 072033R XEC6  STA .LNUM       SEQUENCE NUMBER
C070**
C071***  SET SUCCESSOR STATEMENT  **
C072**
C073  06250 015552R       JSB FLWST
C074  06251 012220R       AND OPMSK     EXTRACT STATEMENT TYPE
C075  06252 001727        ALF,ALF       POSITION
C076  06253 042102R       ADA XECBR     COMPUTE EXECUTION ADDRESS
C077  06254 124000        JMP 0,I       BRANCH TO EXECUTION CODE
C078**
C079***  EVALUATE FORMULA AND RETURN RESULT  **
C080**
C081  06255 000000  FETCH NOP
C082  06256 017723R       JSB FORMX       EVALUATE FORMULA
C083  06257 017266R       JSB OPCHK
C084  06260 036030R       ISZ HSTPT     UNSTACK RESULT ADDRESS
C085  06261 006020        SSB           RESULT ON DISK?
C086  06262 026266R       JMP FTCH2     YES, LOAD A AND B WITH RESULT
C087  06263 104200        DLD 1,I       LOAD RESULT
      06264 100001
C088  06265 126255R       JMP FETCH,I   EXIT
C089  06266 060001  FTCH2 LDA B
C090  06267 017505R       JSB FTHOP      LOAD RESULT
C091  06270 126255R       JMP FETCH,I
C092**
C093***  SET POINTER TO START OF DATA STATEMENT  **
C094**
C095  06271 000000  SETDP NOP           STATEMENT ADDRESS IN (B)
C096  06272 006004        INB           LOAD
C097  06273 017051R       JSB LDBI      STATEMENT LENGTH
C098  06274 003004        CMA,INA       SET
C099  06275 002004        INA             DATA COUNTER
C100  06276 072057R       STA DCCNT         TO 1-STATEMENT LENGTH
C101  06277 006004        INB           SET 'NEXT DATA' POINTER ONE
C102  06300 076036R       STB NXTDT       WORD ABOVE FIRST CONSTANT
C103  06301 126271R       JMP SETDP,I
C104**
C105***  SEARCH FOR STATEMENT OF GIVEN TYPE  **
C106**
C107  06302 000000  STSRH NOP           TYPE IN (A), ADDRESS IN (B)
```

```
C108  06303 012220R        AND OPMSK     EXTRACT
C109  06304 072052R        STA TEMP4       STATEMENT TYPE
C110  06305 060001  STSR1  LDA 1         EXTRACT
C111  06306 040055         ADA .2          PROGRAM
C112  06307 017204R        JSB DREAD                STATEMENT
C113  06310 012220R        AND OPMSK             TYPE
C114  06311 052052R        CPA TEMP4     DESIRED TYPE?
C115  06312 026322R        JMP STSR2     YES
C116  06313 060001         LDA 1         NO, FETCH
C117  06314 002004         INA             STATEMENT LENGTH
C118  06315 017204R        JSB DREAD
C119  06316 044000         ADB A         COMPUTE NEW ADDRESS
C120  06317 056010C        CPB PBPTR     PAST LAST STATEMENT?
C121  06320 126302R        JMP STSRH,I   YES
C122  06321 026305R        JMP STSR1     NO
C123  06322 056302R STSR2  ISZ STSRH
C124  06323 126302R        JMP STSRH,I
C125**
C126***  FETCH A DATA ITEM  **
C127**
C128  06324 000000  FDATA NOP
C129  06325 056037R FDAT1  ISZ DCCNT     MORE DATA?
C130  06326 026335R        JMP FDAT2     YES
C131  06327 162115R        LDA ADATA,I   NO, SEARCH
C132  06330 066036R        LDB NXTDT       FOR NEXT
C133  06331 016302R        JSB STSRH         DATA STATEMENT
C134  06332 016274R        JSB ERROR     NONE FOUND
C135  06333 016271R E4     JSB SETDP     INITIALIZE THE
C136  06334 026325R        JMP FDAT1       DATA POINTERS
C137  06335 056037R FDAT2  ISZ DCCNT     UPDATE
C138  06336 056037R        ISZ DCCNT       POINTER
C139  06337 056036R        ISZ NXTDT
C140  06340 062036R        LDA NXTDT
C141  06341 017204R        JSB DREAD   LOAD
C142  06342 056036R        ISZ NXTDT       DATA
C143  06343 073130R        STA TMP2
C144  06344 062036R        LDA NXTDT
C145  06345 017204R        JSB DREAD
C146  06346 064000         LDB A           ITEM
C147  06347 063130R        LDA TMP2
C148  06350 056036R        ISZ NXTDT     UPDATE POINTER
C149  06351 126324R        JMP FDATA,I
C150**
C151***  SET FOR FOLLOWING STATEMENT  **
C152**
C153  06352 000000  FLWST NOP           (B) HOLDS PRESENT ADDRESS
C154  06353 060001         LDA 1         COMPUTE
C155  06354 002004         INA             ADDRESS
C156  06355 017204R        JSB DREAD          OF
C157  06356 040001         ADA 1               NEXT
C158  06357 072031R        STA PRADD             STATEMENT
C159  06360 017204R        JSB DREAD     RECORD THE
C160  06361 072052R        STA NXTST       SEQUENCE NUMBER
C161  06362 044055         ADB .2        FETCH
C162  06363 076045R        STB TEMP5       FIRST WORD
```

```
C163  06364 017051R        JSB LDBI          OF CURRENT
C164  06365 125552R        JMP FLWST,I          STATEMENT
C165**
C166***  SEARCH STACK FOR GIVEN FOR-VARIABLE  **
C167**
C168  06366 000000  FVSRH NOP
C169  06367 017062R        JSB LDTMP    FETCH
C170  06370 012176R        AND MSK1        FOR-VARIABLE
C171  06371 073440R        STA EDELM     SAVE FOR-VARIABLE
C172  06372 017613R        JSB SSYMT       FIND ADDRESS IN
C173  06373 006004         INB             SYMBOL TABLE
C174  06374 062050R        LDA HSTPT     SAVE
C175  06375 072051R        STA TEMP3       STACK TOP
C176  06376 052010R FVSR1  CPA SYMTF     STACK BOTTOM?
C177  06377 126366R        JMP FVSRH,I   YES, EXIT VIA (P+1)
C178  06400 154000         CPB 0,I       MATCHING FOR-VARIABLE?
C179  06401 026404R        JMP FVSR2     YES
C180  06402 040061         ADA .6        NO, MOVE TO
C181  06403 026376R        JMP FVSR1       NEXT STACK ENTRY
C182  06404 036366R FVSR2  ISZ FVSRH     EXIT
C183  06405 126366R        JMP FVSRH,I     VIA (P+2)
C184*
C185***              ***
C186**  EXECUTE LET  **
C187***              ***
C188*
C189  06406 017723R ELET   JSB FORMX
C190  06407 026233R        JMP XEC4
C191*
C192***                ***
C193**  EXECUTE GO TO  **
C194***                ***
C195*
C196  06410 017604R EGOTO JSB NXSQN    LOAD SEQUENCE NUMBER
C197  06411 026234R        JMP XEC4+1    FIND REFERENCED STATEMENT
C198*
C199***            ***
C200**  EXECUTE IF  **
C201***            ***
C202*
C203  06412 016255R EIF    JSB FETCH      FETCH VALUE OF FORMULA
C204  06413 002003         SZA,RSS       RESULTANT TRUE?
C205  06414 026233R        JMP XEC4      NO
C206  06415 036045R        ISZ TEMPS     YES, BRANCH TO
C207  06416 036045R        ISZ TEMPS
C208  06417 066045R        LDB TEMPS       FOLLOWING
C209  06420 017062R        JSB LDTMP    FOLLOWING STATEMENT
C210  06421 026251R        JMP XSA      AND EXECUTE
C211*
C212***              ***
C213**  EXECUTE FOR  **
C214***              ***
C215*
C216  06422 016366R EFOR   JSB FVSRH     FOR-VARIABLE ALREADY IN STACK?
C217  06423 026434R        JMP EFOR1     NO
```

```
C218  06424 072050R        STA TEMP2     YES, SAVE SOURCE ADDRESS
C219  06425 040061         ADA .6        SAVE
C220  06426 072052R        STA TEMP4       .DESTINATION ADDRESS
C221  06427 075047R        STB TEMP1     SAVE FOR-VARIABLE ADDRESS
C222  06430 015567R        JSB MVTOC       COMPRESS STACK
C223  06431 065047R        LDB TEMP1     RESTORE FOR-VARIABLE ADDRESS
C224  06432 002400         CLA           COMPUTE
C225  06433 002401         CLA,RSS       COMPUTE
C226  06434 060045 EFOR1   LDA M6          NEW TOP OF
C227  06435 042030R        ADA HSTPT         FOR-STACK
C228  06436 072030R        STA HSTPT           POINTER
C229  06437 072047R        STA TEMP1
C230  06440 003004         CMA,INA       STACK
C231  06441 042027R        ADA LSTPT
C232  06442 002021         SSA,RSS         OVERFLOW?
C233  06443 027240R        JMP E1        YES
C234  06444 175047R        STB TEMP1,I   NO, RECORD FOR-VARIABLE ADDRESS
C235  06445 017723R        JSB FORMX      INITIALIZE FOR VARIABLE
C236  06446 035045R        ISZ TEMP5
C237  06447 035047R        ISZ TEMP1     SAVE
C238  06450 062047R        LDA TEMP1       LIMIT
C239  06451 072557R        STA ENEX2         ADDRESS
C240  06452 015255R        JSB FETCH      FETCH
C241  06453 172047R        STA TEMP1,I     AND
C242  06454 035047R        ISZ TEMP1         STORE
C243  06455 175047R        STB TEMP1,I         LIMIT
C244  06456 035047R        ISZ TEMP1
C245  06457 064051         LDB M2        SET FOR STEP SIZE
C246  06460 075524R        STB FDATA       SIGN CHECK
C247  06461 017062R        JSB LDTMP       LOOK FOR
C248  06462 002002         SZA             FOLLOWING ' STEP'
C249  06463 025467R        JMP EFOR2     FOUND
C250  06464 062261R        LDA HONE      NOT FOUND,
C251  06465 064055         LDB .2          DEFAULT
C252  06466 002001         RSS               IS 1.0
C253  06467 015255R EFOR2  JSB FETCH
C254  06470 002020         SSA           STEP SIZE NEGATIVE?
C255  06471 035524R        ISZ FDATA     YES
C256  06472 172047R        STA TEMP1,I   SAVE
C257  06473 035047R        ISZ TEMP1       STEP
C258  06474 175047R        STB TEMP1,I       SIZE
C259  06475 035047R        ISZ TEMP1     SET POINTER
C260  06476 062032R        LDA NXTST       TO STATEMENT
C261  06477 172047R        STA TEMP1,I       FOLLOWING THE FOR
C262  06500 162115R EFOR3  LDA ANEXT,I   FIND
C263  06501 065031R        LDB PRADD       'NEXT'
C264  06502 015502R        JSB STSRH         STATEMENT
C265  06503 000000         NOP
C266  06504 015552R        JSB FLWST     FIND FOLLOWING STATEMENT
C267  06505 012176R        AND MSK1      SAME
C268  06506 053440R        CPA EDELM       FOR-VARIABLE?
C269  06507 002001         RSS           YES
C270  06510 025500R        JMP EFOR3     NO
C271  06511 166030R        LDB HSTPT,I   LOAD
C272  06512 104200         DLD 1,I         VALUE OF FOR VARIABLE
```

```
      06513 100001
C273  06514 026542R          JMP .ENE2    CHECK ACCEPTABILITY
C274*
C275***                  ***
C276**  EXECUTE NEXT     **
C277***                  ***
C278*
C279  06515 016366R ENEX1  JSB FVSRH      FIND CORRESPONDING STACK ENTRY
C280  06516 026233R        JMP XEC4       NONE PRESENT
C281  06517 072030R        STA HSTPT      RESET TOP OF STACK
C282  06520 076556R        STB ENEX1      SAVE FOR-VARIABLE ADDRESS
C283  06521 002004         INA            SAVE LIMIT
C284  06522 072557R        STA ENEX2        ADDRESS
C285  06523 040055         ADA .2         SAVE STEP SIZE
C286  06524 072047R        STA TEMP1        ADDRESS
C287  06525 064051         LDB M2         SET STEP SIZE
C288  06526 076524R        STB FDATA        SIGN CHECK
C289  06527 162047R        LDA TEMP1,I    LOAD
C290  06530 036047R        ISZ TEMP1        STEP
C291  06531 166047R        LDB TEMP1,I        SIZE
C292  06532 036047R        ISZ TEMP1
C293  06533 002020         SSA            CHECK
C294  06534 036324R        ISZ FDATA        SIGN
C295  06535 016002X        FAD ENEX1,I       INCREMENT FOR VARIABLE
      06536 106556R
C296  06537 172556R        STA ENEX1,I        AND
C297  06540 036556R        ISZ ENEX1            SAVE
C298  06541 176556R        STB ENEX1,I            VALUE
C299  06542 016003X .ENE2  FSB ENEX2,I      COMPUTE FOR VARIABLE LIMIT
      06543 106557R
C300  06544 036324R        ISZ FDATA      POSITIVE STEP SIZE?
C301  06545 001600         ELA            YES, COMPLEMENT SIGN
C302  06546 002020         SSA            NO, NON-NEGATIVE RESULT?
C303  06547 026552R        JMP ENEX3      NO
C304  06550 162047R        LDA TEMP1,I    YES, GO TO FIRST
C305  06551 026234R        JMP XEC4+1       STATEMENT OF LOOP
C306  06552 062030R ENEX3  LDA HSTPT      FAILS,
C307  06553 040061         ADA .6           ERASE
C308  06554 072030R        STA HSTPT          STACK
C309  06555 026233R        JMP XEC4             ENTRY
C310  06556 000000  ENEX1  BSS 1
C311  06557 000000  ENEX2  BSS 1
C312*
C313***                  ***
C314**  EXECUTE GOSUB    **
C315***                  ***
C316*
C317  06560 000000  EGOSB  NOP
C318  06561 017604R        JSB NXSQN      LOAD (A) WITH NEXT SEQ NUMBER
C319  06562 066032R        LDB NXTST       LOAD (B) WITH
C320  06563 072032R        STA NXTS1         RETURN SEQUENCE NUMBER
C321  06564 017234R        JSB SLWST      STACK RETURN ON LOW-CORE STACK
C322  06565 040041         ADA M10        GOSUBS NESTED
C323  06566 052012R        CPA LSTAK        10 DEEP?
C324  06567 015274R        JSB ERROR      YES
```

```
C325  06570 026233R E2     JMP XEC4       NO
C326*
C327***                  ***
C328**  EXECUTE RETURN   **
C329***                  ***
C330*
C331  06571 066027R ERTRN LDB LSTPT      RETURN STACK
C332  06572 055012R        CPB LSTAK       EMPTY?
C333  06573 015274R        JSB ERROR      YES
C334  06574 162027R E5     LDA LSTPT,I    NO, LOAD RETURN ADDRESS
C335  06575 044052         ADB M1         RESET
C336  06576 076027R        STB LSTPT        STACK POINTER
C337  06577 026234R        JMP XEC4+1
```

```
C339*
C340***                 ***
C341**  EXECUTE WAIT    **
C342***                 ***
C343*
C344  06600 036045R EWAIT ISZ TEMPS      POINT (TEMPS) TO FORMULA
C345  06601 015255R       JSB FETCH      FETCH EVALUATED FORMULA
C346  06602 002020       SSA            NEGATIVE?
C347  06603 026233R       JMP XEC4       YES
C348  06604 017160R       JSB IFIX       CONVERT TO INTEGER
C349  06605 002404       CLA,INA        LARGE INTEGER
C350  06606 002003       SZA,RSS        SMALL
C351  06607 007021       CMB,SSB,RSS      INTEGER?
C352  06610 065263R       LDB MNEG       NO
C353  06611 006007  EWAI1 INB,SZB,RSS    WAIT?
C354  06612 026233R       JMP XEC4       NO
C355  06613 062254R       LDA M310       YES, SET INNER LOOP
C356  06614 002006       INA,SZA        MORE?
C357  06615 026614R       JMP *-1        YES
C358  06616 026611R       JMP EWAI1      NO
C359*
C360***                 ***
C361**  EXECUTE CALL    **
C362***                 ***
C363*
C364  06617 036045R ECALL ISZ TEMPS      FETCH
C365  06620 036045R       ISZ TEMPS        SUBROUTINE
C366  06621 017062R       JSB LDTMP         NUMBER
C367  06622 064000       LDB A
C368  06623 017115R       JSB FNDSB      FIND
C369  06624 006004       INB              ENTRY
C370  06625 164001       LDB 1,1            POINT AND
C371  06626 076060R ECAL4 STB TEMPS+11
C372  06627 062030R       LDA HSTPT      SAVE HIGH CORE
C373  06630 072057R       STA TEMPS+10     STACK POINTER
C374  06631 036045R ECAL1 ISZ TEMPS      ANY
C375  06632 017062R       JSB LDTMP          PARAMETERS
C376  06633 052202R       CPA B2000   ::
C377  06634 026637R       JMP ECAL2      NO
C378  06635 017723R       JSB FORMX         YES, EVALUATE
C379  06636 025651R       JMP ECAL1       A PARAMETER
C380  06637 003400  ECAL2 CCA            LOAD ADDRESS OF
C381  06640 042057R       ADA TEMPS+10     PARAMETER ADDRESSES
C382  06641 115060R       JSB TEMPS+11,I     AND BRANCH TO SUBROUTINE
C383  06642 062057R       LDA TEMPS+10   RESTORE
C384  06643 072030R       STA HSTPT
C385  06644 065006R       LDB FCORE      POINTERS
C386  06645 076026R       STB TSTPT
C387  06646 026233R       JMP XEC4
```

```
C389*
C390***                  ***
C391**  EXECUTE READ     **
C392***                  ***
C393*
C394  06647 056031R EREAD CPB PRADD     END-OF-STATEMENT?
C395  06650 025233R       JMP XEC4      YES
C396  06651 017725R       JSB FORMX      NO, EVALUATE THE NEXT RECORD
C397  06652 162030R       LDA HSTPT,I   RECORD
C398  06653 073461R       STA OUTLN       ADDRESS
C399  06654 002020       SSA
C400  06655 025665R       JMP ERD2
C401  06656 015524R       JSB FDATA     GET DATA ITEM
C402  06657 104400       DST OUTLN,I   STORE DATA ITEM
      06660 101461R
C403  06661 056030R ERD1  ISZ HSTPT
C404  06662 065045R       LDB TEMPS
C405  06663 006004       INB
C406  06664 025647R       JMP EREAD
C407  06665 015524R ERD2  JSB FDATA
C408  06666 017342R       JSB ODST
C409  06667 001461R       DEF OUTLN
C410  06670 025661R       JMP ERD1
C411**
C412***  INITIALIZE FOR PRINT  **
C413**
C414  06671 000000  PRNIN NOP
C415  06672 003400       CCA           INITIALIZE
C416  06673 042016R       ADA .BUFA       BUFFER
C417  06674 072017R       STA BADDR         POINTER
C418  06675 062034R       LDA TYPE      INITIALIZE
C419  06676 003004       CMA,INA         'CHARACTERS OUTPUTTED'
C420  06677 072020R       STA CCNT          COUNTER
C421  06700 002011       SLA,RSS       START ON ODD CHARACTER POSITION?
C422  06701 125671R       JMP PRNIN,I   NO
C423  06702 040052       ADA M1        YES, BIAS
C424  06703 072020R       STA CCNT        COUNTER
C425  06704 002400       CLA           OUTPUT A
C426  06705 017477R       JSB OUTCR       NULL CHARACTER
C427  06706 125671R       JMP PRNIN,I
C428*
C429***                  ***
C430**  EXECUTE PRINT    **
C431***                  ***
C432*
C433  06707 002400  EPRIN CLA
C434  06710 072272R       STA PARM
C435  06711 015671R       JSB PRNIN
C436  06712 002400       CLA           TURN ON
C437  06713 025720R       JMP EPRI1+1     'END-OF-LINE' FLAG
C438  06714 002400  EPRI0 CLA           EXECUTE COMMA
C439  06715 052402R       CPA EOL         IF NOT FOLLOWING
C440  06716 017440R       JSB EDELM         A TAB
C441  06717 003400  EPRI1 CCA           TURN OFF
C442  06720 072402R       STA EOL         'END-OF-LINE' FLAG
```

```
C443  06721 017062R        JSB LDTMP     EXTRACT
C444  06722 012225R        AND OPDMK       OPERAND
C445  06723 002002         SZA           NULL?
C446  06724 026744R        JMP EPRI3     NO, FORMULA OR TAB
C447  06725 036045R EPRI2  ISZ TEMPS     YES
C448  06726 066045R        LDB TEMPS     MORE
C449  06727 056031R        CPB PRADD       STATEMENT?
C450  06730 027010R        JMP EPRI7     NO
C451  06731 017062R        JSB LDTMP   YES, EXTRACT
C452  06732 012220R        AND OPMSK       NEXT OPERATOR
C453  06733 052177R        CPA B1000   ::
C454  06734 026714R        JMP EPRI0     YES
C455  06735 052200R        CPA B1400   ::
C456  06736 026717R        JMP EPRI1     YES
C457  06737 052174R        CPA B400
C458  06740 026753R        JMP EPRI4     YES
C459  06741 003400         CCA           NO, MUST BE +,-, OR (
C460  06742 042045R        ADA TEMPS     BACKUP TO PRIOR
C461  06743 072045R        STA TEMPS       NULL OPERAND
C462  06744 003400  EPRI3  CCA           SET
C463  06745 072402R        STA EOL         TAB FLAG
C464  06746 016255R        JSB FETCH     EVALUATE
C465  06747 036402R        ISZ EOL       TAB?
C466  06750 026725R        JMP EPRI2     YES
C467  06751 017426R        JSB ENOUT     NO, PRINT NUMBER
C468  06752 026725R        JMP EPRI2
C469  06753 002400  EPRI4  CLA           TURN ON
C470  06754 072402R        STA EOL         'END-OF-LINE' FLAG
C471  06755 073254R        STA SLWST     ZERO
C472  06756 073461R        STA OUTLN       CHARACTER COUNT
C473  06757 017051R EPRI5  JSB LDBI
C474  06760 010074         AND MSK0      NON-NULL
C475  06761 002003         SZA,RSS         LOW CHARACTER?
C476  06762 026773R        JMP EPRI6     NO
C477  06763 037461R        ISZ OUTLN     YES, COUNT IT
C478  06764 006004         INB
C479  06765 017051R        JSB LDBI
C480  06766 012220R        AND OPMSK
C481  06767 052174R        CPA B400    ::
C482  06770 026773R        JMP EPRI6     YES
C483  06771 037461R        ISZ OUTLN     NO, COUNT HIGH CHARACTER
C484  06772 026757R        JMP EPRI5
C485  06773 067461R EPRI6  LDB OUTLN     WILL
C486  06774 046020R        ADB CCNT        LINE
C487  06775 062272R        LDA PARM
C488  06776 002021         SSA,RSS
C489  06777 027003R        JMP *+4
C490  07000 046237R        ADB M21
C491  07001 006021         SSB,RSS
C492  07002 016274R .WRER   JSB ERROR
C493  07003 046250R WRERR  ADB M73
C494  07004 006021         SSB,RSS             72 CHARACTERS?
C495  07005 017461R        JSB OUTLN     YES, GET FRESH LINE
C496  07006 017172R        JSB OUTST       OUTPUT STRING
C497  07007 026721R        JMP EPRI1+2
```

```
C498  07010 036402R EPRI7 ISZ EOL        'END-OF-LINE' ?
C499  07011 027034R       JMP EPRI8      YES
C500  07012 066034R       LDB TYPE       NO, LOAD COUNT OF
C501  07013 007004        CMB,INB          CHARACTERS OUTPUTTED
C502  07014 076272R       STB PARM
C503  07015 062171R       LDA B137
C504  07016 017477R       JSB OUTCR
C505  07017 066272R       LDB PARM
C506  07020 062020R       LDA CCNT       LOAD LINE LENGTH
C507  07021 003004        CMA,INA        SAVE NEW COUNT OF
C508  07022 072034R       STA TYPE         CHARACTERS OUTPUTTED
C509  07023 040001        ADA 1          COMPUTE CHARACTERS NOT YET OUT
C510  07024 004010        SLB            CORRECT FOR START ON
C511  07025 040052        ADA M1           ODD PRINT POSITION
C512  07026 003004        CMA,INA
C513  07027 066016R       LDB .BUFA      OUTPUT
C514  07030 002002        SZA              NON-EMPTY
C515  07031 017161R       JSB EXECW            BUFFER
C516  07032 036034R       ISZ TYPE
C517  07033 025233R       JMP XEC4
C518  07034 017461R EPRI8 JSB OUTLN      PRINT LINE
C519  07035 025233R       JMP XEC4
C520**
C521***  TAB TELEPRINTER  **
C522**
C523  07036 117062R ETAB  JSB IENTA,I    SMALL INTEGER?
C524  07037 027060R       JMP ETAB1      NO
C525  07040 006400        CLB            YES, SET
C526  07041 076402R       STB EOL          'TAB' FLAG TRUE
C527  07042 042247R       ADA M72        EXCEED
C528  07043 002021        SSA,RSS          72?
C529  07044 027060R       JMP ETAB1      YES
C530  07045 003004        CMA,INA        NO, COMPUTE
C531  07046 042247R       ADA M72          BLANKS?
C532  07047 042020R       ADA CCNT           REQUIRED
C533  07050 002021        SSA,RSS        ANY?
C534  07051 026101R       JMP FOR12       NO
C535  07052 073461R       STA OUTLN      YES,
C536  07053 062141R       LDA .32          OUTPUT
C537  07054 017477R       JSB OUTCR          REQUIRED
C538  07055 037461R       ISZ OUTLN            NUMBER
C539  07056 027053R       JMP *-3                 OF BLANKS
C540  07057 026101R       JMP FOR12
C541  07060 017461R ETAB1 JSB OUTLN      OUTPUT THE
C542  07061 026101R       JMP FOR12          LINE
C543*
C544  07062 011420R IENTA DEF .IENT
C545*
C546***                 ***
C547**  EXECUTE INPUT  **
C548***                 ***
C549*
C550  07063 002006  EINP1 INA,SZA        END-OF-INPUT?
C551  07064 017667R       JSB DRGST      YES, CALL FOR MORE
C552  07065 016402R EINP2 JSB CONST      CONVERT AND STORE NUMBER
```

```
C553  07066  027063R        JMP EINP1      NOT NUMBER
C554  07067  066045R        LDB TEMPS      END-OF-
C555  07070  006004         INB
C556  07071  056031R        CPB PRADD        STATEMENT?
C557  07072  027106R        JMP EINP3      YES
C558  07073  050065         CPA .10        NO, INSURE MORE INPUT
C559  07074  017667R EINPT  JSB DRQST       CALL FOR INPUT
C560  07075  017723R        JSB FORMX      COMPUTE THE VARIABLE ADDRESS
C561  07076  162030R        LDA HSTPT,I
C562  07077  002020         SSA
C563  07100  027103R        JMP *+3
C564  07101  003400         CCA            STORE
C565  07102  142030R        ADA HSTPT,I      ADDRESS-1
C566  07103  056030R        ISZ HSTPT          IN
C567  07104  072022R        STA SBPTR            POINTER
C568  07105  027065R        JMP EINP2
C569  07106  002400  EINP3  CLA            RESET
C570  07107  072034R        STA TYPE         OUTPUT BUFFER
C571  07110  026233R        JMP XEC4
```

```
C573**
C574***  EXIT FORMULA ON EMPTY STACK  **
C575**
C576  07111 105723R        DEF FORMX,I
C577*
C578***                   ***
C579**  EXECUTE RESTORE   **
C580***                   ***
C581*
C582  07112 066035R ERSTR LDB DSTRT     GET FIRST DATA STATEMENT ADDRESS
C583  07113 054052         CPB M1       IMPOSSIBLE ADDRESS?
C584  07114 026233R        JMP XEC4     YES, DONE
C585  07115 016271R        JSB SETDP    NO, SET DATA POINTER
C586  07116 026233R        JMP XEC4     DONE
C587**
C588***  FORMULA OPERATOR JUMP TABLE  **
C589**
C590  07117 007164R AROTB DEF ESCMA     SUBSCRIPT SEPARATOR
C591  07120 007251R        DEF ESTR     ASSIGNMENT OPERATOR
C592  07121 007344R        DEF EFAD     '+'
C593  07122 007350R        DEF EFSB     '-'
C594  07123 007354R        DEF EFMP     '*'
C595  07124 007360R        DEF EFDV     '/'
C596  07125 007364R        DEF EPWR     '^'
C597  07126 007516R        DEF EGTRT    '>'
C598  07127 007524R        DEF ELST     '<'
C599  07130 007554R        DEF ENEQL    '#'
C600  07131 007532R        DEF EEQL     '='
C601  07132 007567R        DEF EUMIN    UNARY '-'
C602  07133 007572R        DEF ELBRC    '['
C603  07134 005726R        DEF FORM1     '('
C604  07135 005776R        DEF FOR11     UNARY
C605  07136 007153R        DEF EOR      OR
C606  07137 007160R        DEF EAND     AND
C607  07140 007600R        DEF ENOT     NOT
C608  07141 007540R        DEF EGORE    '>='
C609  07142 007546R        DEF ELORE    '<='
C610**
C611***  EXECUTE A BINARY OPERATOR  **
C612**
C613  07143 000000  BINOP NOP          SAVE
C614  07144 017266R       JSB OPCHK    SAVE ADDRESS OF
C615  07145 077152R       STB BIN02      TOP OPERAND
C616  07146 036030R       ISZ HSTPT    UNSTACK ADDRESS
C617  07147 017252R       JSB STTOP    LOAD SECOND OPERAND
C618  07150 127143R       JMP BINOP,I
C619  07151 000000  BIN01 BSS 1
C620  07152 000000  BIN02 BSS 1
C621**
C622***  EXECUTE OR  **
C623**
C624  07153 017143R EOR   JSB BINOP    VALIDATE
C625  07154 002002  ORS   SZA          SECOND OPERAND NON-ZERO?
C626  07155 027564R       JMP TRUE     YES
C627  07156 017517R ORS1  JSB LDOP2     NO, CHECK SECOND
```

```
C628  07157 027557R       JMP ENEQ1     OPERAND
C629**
C630***  EXECUTE AND **
C631**
C632  07160 017143R EAND  JSB BINOP     VALIDATE
C633  07161 002003  ANDS  SZA,RSS       SECOND OPERAND ZERO?
C634  07162 027561R       JMP FALSE     YES
C635  07163 027156R       JMP URS1      NO
```

```
C637**
C638***  EXECUTE SUBSCRIPT COMMA  **
C639**
C640   07164 017234R ESCMA JSB ESBS      INTEGERIZE COLUMN SUBSCRIPT
C641   07165 035027R       ISZ LSTPT
C642   07166 017234R       JSB ESBS      INTEGERIZE ROW SUBSCRIPT
C643   07167 166030R       LDB HSTPT,I   FETCH
C644   07170 044055        ADB .2          SUBSCRIPT
C645   07171 160001        LDA 1,I           BOUNDS
C646   07172 010074        AND MSK0      SAVE
C647   07173 073461R       STA OUTLN       COLUMN BOUND
C648   07174 160001        LDA 1,I       EXTRACT
C649   07175 001727        ALF,ALF         ROW
C650   07176 010074        AND MSK0          BOUND
C651   07177 003004        CMA,INA       ACTUAL
C652   07200 142027R       ADA LSTPT,I     ROW SUBSCRIPT
C653   07201 002021        SSA,RSS           LEGAL?
C654   07202 027221R       JMP E6-1      NO
C655   07203 063461R       LDA OUTLN     YES
C656   07204 050054        CPA .1        COLUMN MATRIX?
C657   07205 027211R       JMP ESCM1     YES
C658   07206 100200        MPY LSTPT,I     NO, COMPUTE ADDRESS DISPLACEMENT
       07207 100027R
C659   07210 002001        RSS               DUE TO ROWS
C660   07211 162027R ESCM1 LDA LSTPT,I
C661   07212 007400        CCB           UNSTACK
C662   07213 045027R       ADB LSTPT       ROW
C663   07214 075027R       STB LSTPT         SUBSCRIPT
C664   07215 067461R       LDB OUTLN     ACTUAL
C665   07216 007004        CMB,INB         COLUMN
C666   07217 145027R       ADB LSTPT,I       SUBSCRIPT
C667   07220 006021        SSB,RSS             LEGAL?
C668   07221 016274R       JSB ERROR     NO
C669   07222 142027R E6    ADA LSTPT,I   YES, ADD IN COLUMN DISPLACEMENT
C670   07223 001000        ALS           DOUBLE DISPLACEMENT
C671   07224 166030R       LDB HSTPT,I   COMPUTE
C672   07225 140001        ADA 1,I         ACTUAL
C673   07226 052265R       IOR FLGBT           ADD V.A. BIT
C674   07227 172030R       STA HSTPT,I       ADDRESS
C675   07230 066027R       LDB LSTPT     UNSTACK
C676   07231 044052        ADB M1
C677   07232 075027R       STB LSTPT           [
C678   07233 027726R       JMP FORM1
C679**
C680***  INTEGERIZE A SUBSCRIPT  **
C681**
C682   07234 000000 ESBS   NOP
C683   07235 017266R       JSB OPCHK     VALIDATE SUBSCRIPT
C684   07236 006021        SSB,RSS         SUBSCRIPT ON DISK?
C685   07237 027243R       JMP ESB1        NO, FETCH SUBSCRIPT
C686   07240 060001        LDA B
C687   07241 017505R       JSB FTHOP       YES, FETCH FROM DISK
C688   07242 027245R       JMP *+3
C689   07243 104200 ESB1   DLD 1,I
       07244 100001
```

```
C690  07245 017147R       JSB SBFIX      INTEGERIZE
C691  07246 175027R       STB LSTPT,I    SAVE IN OPERATOR STACK
C692  07247 036030R       ISZ HSTPT      POP OPERAND STACK
C693  07250 127234R       JMP ESBS,I
```

```
C695**
C696***  EXECUTE STORE  **
C697**
C698  07251 066054R ESTR  LDB TEMPS+7    IS NEXT OPERATOR
C699  07252 006002        SZB              AN END-OF-FORMULA?
C700  07253 026005R        JMP FOR10       NO, DEFER AND STORE
C701  07254 056053R        CPB TEMPS+6    YES, FIRST STORE OPERATOR USED?
C702  07255 027305R        JMP ESTR2      YES
C703  07256 162030R ESTR1  LDA HSTPT,I    SET
C704  07257 072056R        STA TEMPS+9      .DESTINATION
C705  07260 062053R        LDA TEMPS+6    SOURCE ADDRESS IN (A)
C706  07261 002020        SSA         SOURCE ADDR. ON DISK?
C707  07262 027311R        JMP ESTR4    YES, TRANSFER DATA
C708  07263 164000        LDB 0,I        TRANSFER HIGH
C709  07264 073130R        STA TMP2      SAVE SOURCE ADDR.
C710  07265 062056R        LDA TEMPS+9  .DESTINATION ADDR ON DISK?
C711  07266 002020        SSA
C712  07267 017334R        JSB ESTR5     STORE DATA & RETURN (P+2)
C713  07270 176056R        STB TEMPS+9,I    PART OF SOURCE
C714  07271 063130R        LDA TMP2
C715  07272 036056R        ISZ TEMPS+9    UPDATE
C716  07273 002004        INA              POINTERS
C717  07274 164000        LDB 0,I        TRANSFER LOW
C718  07275 073130R        STA TMP2      SAVE SOURCE ADDR.
C719  07276 062056R        LDA TEMPS+9  .DESTINATION ADDR. ON DISK?
C720  07277 002020        SSA
C721  07300 017334R        JSB ESTR5     YES, STORE DATA & RETURN (P+2)
C722  07301 176056R        STB TEMPS+9,I    PART OF SOURCE
C723  07302 063130R        LDA TMP2
C724  07303 036030R        ISZ HSTPT      POP STACK
C725  07304 027776R        JMP FOR11
C726  07305 017266R ESTR2  JSB OPCHK      SAVE ADDRESS
C727  07306 076053R        STB TEMPS+6      OF QUANTITY
C728  07307 036030R        ISZ HSTPT      YES, POP HIGH-CORE
C729  07310 027256R        JMP ESTR1        STACK AND EXECUTE STORE
C730  07311 012222R ESTR4  AND INF      REMOVE V.A. BIT
C731  07312 073130R        STA TMP2     SAVE SOURCE ADDRESS
C732  07313 017204R        JSB DREAD     HIGH PART OF SOURCE--> (A)
C733  07314 070001        STA B
C734  07315 062056R        LDA TEMPS+9      .DESTINATION ON DISK?
C735  07316 002020        SSA
C736  07317 017334R        JSB ESTR5        YES, STORE LOW PART OF SOURCE
C737  07320 176056R        STB TEMPS+9,I
C738  07321 036056R        ISZ TEMPS+9    UPDATE POINTERS
C739  07322 037130R        ISZ TMP2
C740  07323 063130R        LDA TMP2    GET SOURCE ADDR.
C741  07324 017204R        JSB DREAD   LOW PART OF SOURCE --> (A)
C742  07325 070001        STA B
C743  07326 062056R        LDA TEMPS+9      .DESTINATION ON DISK?
C744  07327 002020        SSA
C745  07330 017334R        JSB ESTR5       YES, STORE HIGH PART OF SOURCE
C746  07331 176056R        STB TEMPS+9,I
C747  07332 036030R        ISZ HSTPT    POP HIGH CORE
C748  07333 027776R        JMP FOR11
C749  07334 000000  ESTR5  NOP
```

```
C750  07355 012222R        AND INF          REMOVE V.A. BIT
C751  07356 073132R        STA TMP3         AND SAVE
C001  07357 060001         LDA B
C002  07360 017151R        JSB DSTOR        (A)--> V.A. ON DISK
C003  07361 015132R        DEF TMP3
C004  07362 037334R        ISZ ESTR5
C005  07363 127334R        JMP ESTR5,I
C006**
C007***  CALL ADD  **
C008**
C009  07364 017143R EFAD   JSB BINOP
C010  07365 017403R        JSB .FAD.
C011  07366 007152R        DEF BIN02
C012  07367 027771R        JMP FORM0
C013**
C014***  CALL SUBTRACT  **
C015**
C016  07370 017143R EFSB   JSB BINOP
C017  07371 017412R        JSB .FSB.
C018  07372 007152R        DEF BIN02
C019  07373 027771R        JMP FORM0
C020**
C021***  CALL MULTIPLY  **
C022**
C023  07374 017143R EFMP   JSB BINOP
C024  07375 017421R        JSB .FMP.
C025  07376 007152R        DEF BIN02
C026  07377 027771R        JMP FORM0
C027**
C028***  CALL DIVIDE  **
C029**
C030  07400 017143R EFDV   JSB BINOP
C031  07401 017430R        JSB .FDV.
C032  07402 007152R        DEF BIN02
C033  07403 027771R        JMP FORM0
C034**
C035***  EXECUTE ^  **
C036**
C037  07404 166030R EPWR   LDB HSIPT,I   LOAD
C038  07405 077130R        STB TMP2
C039  07406 017364R        JSB F/PAR          POWER
C040  07407 015130R        DEF TMP2
C041  07410 017160R        JSB IFIX
C042  07411 027574R        JMP *+3
C043  07412 102301         SOS              INTEGER?
C044  07413 027406R        JMP EPWR1        YES
C045  07414 017143R        JSB BINOP        NO
C046  07415 017464R RPWR   JSB PCHK         CHECK ARGUMENTS
C047  07416 002020         SSA              NEGATIVE BASE?
C048  07417 016274R        JSB ERROR        YES
C049  07420          BASER EQU *
C050  07420 067151R        LDB BIN01        NO, LOAD BASE
C051  07421 115067R        JSB .LOGA,I      TAKE NATURAL LOG
C052  07422 017421R        JSB .FMP.      MULTIPLY BY POWER
C053  07423 007152R        DEF BIN02
```

```
C054  07404 017206R        JSB .EXP        EXPONENTIATE
C055  07405 027771R        JMP FORM0          RESULT
C056  07406 077463R EPWR1  STB TT1         SAVE SIGN
C057  07407 006020         SSB             SAVE
C058  07410 007004         CMB,INB           ABSOLUTE VALUE
C059  07411 077464R        STB TT2             OF POWER
C060  07412 017143R        JSB BINOP
C061  07413 017464R IPWR   JSB PCHK        CHECK ARGUMENTS
C062  07414 067151R        LDB BIN01       STORE
C063  07415 073151R        STA BIN01
C064  07416 077152R        STB BIN02         BASE
C065  07417 062261R        LDA HONE        INITIALIZE
C066  07420 072051R        STA TT3           RESULT
C067  07421 060055         LDA .2              TO
C068  07422 072052R        STA TT4               1.0
C069  07423 067464R IPWR1  LDB TT2         DIVIDE POWER
C070  07424 004031         SLB,BRS           BY 2
C071  07425 027444R        JMP IPWR3       WAS ODD
C072  07426 077464R        STB TT2         WAS EVEN
C073  07427 006002  IPWR2  SZB             ZERO?
C074  07430 027455R        JMP IPWR4       NO
C075  07431 063463R        LDA TT1         YES
C076  07432 002020         SSA             POSITIVE POWER?
C077  07433 027437R        JMP IPWR5       NO
C078  07434 062051R        LDA TT3         YES,LOAD
C079  07435 066052R        LDB TT4           RESULT
C080  07436 027771R        JMP FORM0
C081  07437 062261R IPWR5  LDA HONE        LOAD
C082  07440 064055         LDB .2            1.0
C083  07441 015004X        JSB .FDV        DIVIDE BY
C084  07442 000051R        DEF TT3           RESULT
C085  07443 027771R        JMP FORM0
C086  07444 077464R IPWR3  STB TT2         SAVE POWER
C087  07445 063151R        LDA BIN01       LOAD
C088  07446 067152R        LDB BIN02         BASE
C089  07447 015001X        FMP TT3             MULTIPLY BY RESULT-SO-FAR
      07450 000051R
C090  07451 072051R        STA TT3         SAVE PARTIAL
C091  07452 075052R        STB TT4           RESULT
C092  07453 067464R        LDB TT2         LOAD POWER
C093  07454 027427R        JMP IPWR2
C094  07455 063151R IPWR4  LDA BIN01       LOAD
C095  07456 067152R        LDB BIN02         BASE
C096  07457 015001X        FMP BIN01           SQUARE IT
      07460 007151R
C097  07461 073151R        STA BIN01       SAVE
C098  07462 077152R        STB BIN02         RESULT
C099  07463 027423R        JMP IPWR1
C100**
C101***  INSURE VALID OPERATION  **
C102**
C103  07464 000000  PCHK   NOP
C104  07465 077151R        STB BIN01       LOAD
C105  07466 073130R        STA IMP2
C106  07467 063152R        LDA BIN02
```

```
C107  07470 002020        SSA             POWER ON DISK?
C108  07471 027475R       JMP PCK1        YES
C109  07472 164000        LDB A,I         NO,  LOAD POWER FROM CORE
C110  07473 063130R       LDA TMP2
C111  07474 027500R       JMP *+4
C112  07475 012222R PCK1  AND INF
C113  07476 017204R       JSB DREAD    LOAD POWER FROM DISK
C114  07477 064000        LDB A
C115  07500 063130R       LDA TMP2
C116  07501 002002        SZA             BASE ZERO?
C117  07502 027513R       JMP PCHK1       NO
C118  07503 006003        SZB,RSS         YES, POWER ZERO?
C119  07504 015274R       JSB ERROR       YES
C120  07505          POWER EQU *
C121  07505 006021        SSB,RSS         NO, POWER POSITIVE?
C122  07506 027561R       JMP FALSE       YES
C123  07507 015274R       JSB ERROR       NO
C124  07510 062222R ZRTNG LDA INF         USE POSITIVE
C125  07511 064051        LDB M2            INFINITY
C126  07512 027771R       JMP FORM0
C127  07513 006003  PCHK1 SZB,RSS         POWER ZERO?
C128  07514 027564R       JMP TRUE        YES, RETURN 1.0
C129  07515 127464R       JMP PCHK,I      NO
C130**
C131***  EXECUTE >  **
C132**
C133  07516 017143R EGTRT JSB BINOP      COMPUTE OPERAND
C134  07517 017412R       JSB .FSB.
C135  07520 007152R       DEF BINO2
C136  07521 002020        SSA             NEGATIVE?
C137  07522 027561R       JMP FALSE       YES
C138  07523 027557R       JMP ENEQ1       NO
C139**
C140***  EXECUTE <  **
C141**
C142  07524 017143R ELST  JSB BINOP      COMPUTE OPERAND
C143  07525 017412R       JSB .FSB.
C144  07526 007152R       DEF BINO2
C145  07527 002020        SSA             NEGATIVE?
C146  07530 027564R       JMP TRUE        YES
C147  07531 027561R       JMP FALSE       NO
C148**
C149***  EXECUTE =  **
C150**
C151  07532 017143R EEQL  JSB BINOP      COMPUTE OPERAND
C152  07533 017412R       JSB .FSB.
C153  07534 007152R       DEF BINO2
C154  07535 002002  EEQL1 SZA             ZERO?
C155  07536 027561R       JMP FALSE       NO
C156  07537 027564R       JMP TRUE        YES
```

```
C158**
C159***  EXECUTE >=  **
C160**
C161  07540 017143R EGORE JSB BINOP     COMPUTE OPERAND
C162  07541 017412R       JSB .FSB.
C163  07542 007152R       DEF BIN02
C164  07543 002020        SSA           POSITIVE?
C165  07544 027561R       JMP FALSE     NO
C166  07545 027564R       JMP TRUE      YES
C167**
C168***  EXECUTE <=  **
C169**
C170  07546 017143R ELORE JSB BINOP     COMPUTE OPERAND
C171  07547 017412R       JSB .FSB.
C172  07550 007152R       DEF BIN02
C173  07551 002020        SSA           NEGATIVE?
C174  07552 027564R       JMP TRUE      YES
C175  07553 027535R       JMP EEQL1     NO
C176**
C177***  EXECUTE #  **
C178**
C179  07554 017143R ENEQL JSB BINOP     COMPUTE OPERAND
C180  07555 017412R       JSB .FSB.
C181  07556 007152R       DEF BIN02
C182  07557 002002  ENEQ1 SZA           NON-ZERO?
C183  07560 027564R       JMP TRUE      YES
C184**
C185***  SET LOGICAL VALUES  **
C186**
C187  07561 002400  FALSE CLA           LOAD
C188  07562 006400        CLB             ZERO
C189  07563 027771R       JMP FORM0
C190  07564 062261R TRUE  LDA ONE       LOAD
C191  07565 064055        LDB .2          ONE
C192  07566 027771R       JMP FORM0
C193**
C194***  EXECUTE UNARY -  **
C195**
C196  07567 017252R EUMIN JSB STTOP     LOAD NUMBER
C197  07570 017217R       JSB ARINV     NEGATE NUMBER
C198  07571 027771R       JMP FORM0
C199**
C200***  EXECUTE LEFT BRACKET  **
C201**
C202  07572 055027R ELBRC ISZ LSTPT     LOAD SUBSCRIPT COMMA
C203  07573 066201R       LDB SCCNT       INFORMATION WORD
C204  07574 017234R       JSB SLWST     STACK IT
C205  07575 017243R       JSB BHSTP     STACK
C206  07576 017321R       JSB RSCHK
C207  07577 027564R       JMP TRUE        1
C208**
C209***  EXECUTE NOT  **
C210**
C211  07600 017252R ENOT  JSB STTOP     LOAD OPERAND
C212  07601 002002        SZA           ZERO?
```

```
C213  07602 027561R        JMP FALSE      NO
C214  07603 027564R        JMP TRUE       YES
C215*
C216  07604 000000  NXSQN  NOP             EVALUATE FORMULA AND FIND
C217  07605 076045R        STB TEMPS      NEXT SEQUENCE NUMBER
C218  07606 016255R        JSB FETCH
C219  07607 017160R        JSB IFIX       INTEGERIZE SEQUENCE
C220  07610 000000        NOP            NUMBER
C221  07611 060001        LDA B
C222  07612 127604R        JMP NXSQN,I
C223*
C224  06251          XSA   EQU XEC6+2
```

```
C226*                ******************************
C227*                SYMBOL TABLE SEARCH SUBROUTINE
C228*                ******************************
C229*
C230*                THE SUBROUTINE IS CALLED WITH THE IDENTIFIER TO BE
C231*                SEARCHED FOR IN A . THE SUBROUTINE RETURNS WITH
C232*                THE ADDRESS OF THE MATCHING ENTRY IN B OR -1 IN
C233*                B IF THERE IS NO MATCHING ENTRY
C234*                THE FOLLOWING RULES APPLY WHEN SEARCHING FOR ARRAYS
C235*
C236*                TYPE 1 (1 DIMENSION) SEARCH FOR CORRESPONDING TYPE 1
C237*                OR TYPE 3 ARRAY. IF TYPE 3 IS FOUND CHANGE THE ENTRY
C238*                TYPE TO TYPE 1
C239*
C240*                TYPE 2 (2 DIMENSIONS) SEARCH FOR CORRESPONDING TYPES
C241*                OR TYPE 3 ARRAY. IF TYPE 3 IS FORND CHANGE THE ENTRY
C242*                TYPE TO TYPE 2
C243*
C244*                TYPE 3 (UNDIMENSIONED) SEARCH FOR CORRESPONDING
C245*                TYPE 3 OR TYPE 1 OR TYPE 2 ARRAY
C246*
C247  07613 000000  SSYMT NOP
C248  07614 072051R       STA STEMP     STORE IDENTIFIER
C249  07615 010062       AND .7     ISOLATE IDENTIFIER TYPE ::
C250  07616 040047       ADA M4
C251  07617 002024       SSA,INA
C252  07620 027624R      JMP *+4       JUMP IF ARRAY TYPE
C253  07621 062051R      LDA STEMP     RESTORE A
C254  07622 070001       STA 1         STORE IN B
C255  07623 027636R      JMP SYMT1+3
C256  07624 002020       SSA           SKIP IF UNDIMENSIONED
C257  07625 027633R      JMP SYMT1
C258  07626 062051R      LDA STEMP     RESTORE A
C259  07627 010044       AND MSK3      177771B SET TYPE TO 1
C260  07630 070001       STA 1
C261  07631 006004       INB           SET TYPE IN B TO 2
C262  07632 027636R      JMP *+4
C263  07633 007400  SYMT1 CCB           SET DIMENSIONED FLAG IN B
C264  07634 060056       LDA .3
C265  07635 052051R      IOR STEMP     SET TYPE TO UNDEFINED
C266  07636 072052R      STA STEMP+1   STORE A
C267  07637 076053R      STB STEMP+2   STORE B
C268  07640 066010R      LDB SYMTF     START OF SYMBOL TABLE
C269  07641 027662R      JMP SYMT4
C270  07642 160401  SYMT2 LDA 1,I       PICK UP 1ST WORD OF ENTRY
C271  07643 052051R      CPA STEMP     COMPARE WITH IDENTIFIER
C272  07644 127615R      JMP SSYMT,I   MATCH ?  RETURN
C273  07645 052052R      CPA STEMP+1   COMPARE WITH DIFFERENT DIM.
C274  07646 027667R      JMP SYMT3
C275  07647 052053R      CPA STEMP+2   COMPARE WITH DIFFERENT DIM.
C276  07650 027667R      JMP SYMT3
C277  07651 160001       LDA 1,I
C278  07652 010062       AND .7     ISOLATE ENTRY TYPE  ::
C279  07653 050062       CPA .7      FUNCTION
C280  07654 027661R      JMP *+5       YES
```

```
C281  07655 040047          ADA M4
C282  07656 002020          SSA             ARRAY ?
C283  07657 005004          INB             YES INCREMENT POINTER
C284  07660 005004          INB             INCREMENT POINTER
C285  07661 044055          ADB .2          ADD 2 TO POINTER
C286  07662 056011R SYMT4   CPB SYMTA        SYMBOL TABLE EXHAUSTED?
C287  07663 007401          CCB,RSS         YES
C288  07664 027642R          JMP SYMT2       NO, CHECK NEXT ENTRY FOR MATCH
C289  07665 062051R          LDA STEMP       RETRIEVE SYMBOL
C290  07666 127613R          JMP SSYMT,I     RETURN WITH B NEGATIVE
C291  07667 062051R SYMT3    LDA STEMP       RESTORE A
C292  07670 036053R          ISZ STEMP+2     DIMENSIONED IDENTIFIER?
C293  07671 002001          RSS             NO, SKIP
C294  07672 170001          STA 1,I         YES CHANGE 1ST WORD OF ENTRY TO
C295  07673 127613R          JMP SSYMT,I     APPROPRIATE DIMENSION TYPE
```

```
C297**
C298***  ERROR TABLE  **
C299**
C300  07674 000071R ERR    DEF EOF+1       PREMATURE STATEMENT END
C301  07675 001643R         DEF RTLE        INPUT EXCEEDS 71 CHARACTERS
C302  07676 001705R         DEF INVSC       SYSTEM COMMAND NOT RECOGNIZED
C303  07677 002202R         DEF SYNE1       NO STATEMENT TYPE FOUND
C304  07700 000532R         DEF NUMER+1     BAD EXPONENT PART
C305  07701 002662R         DEF SYE16       NO LETTER WHERE EXPECTED
C306  07702 002317R         DEF SYNE2       LET STATEMENT HAS NO STORE
C307  07703 002332R         DEF SYNE3       ILLEGAL COM STATEMENT
C308  07704 002357R         DEF SYNE4+1     NO FUNCTION IDENTIFIER (OR BAD)
C309  07705 002371R         DEF SYNE5       MISSING PARAMETER
C310  07706 002376R         DEF SYNE6+1     MISSING ASSIGNMENT OPERATOR
C311  07707 002423R         DEF SYNE7       MISSING 'THEN'
C312  07710 002447R         DEF SYNE8+1     MISSING OR IMPROPER FOR-VARIABLE
C313  07711 002461R         DEF SYNE9       MISSING 'TO'
C314  07712 002474R         DEF SYE10       BAD 'STEP' PART IN FOR STATEMENT
C315  07713 001122R         DEF CALER       CALLED ROUTINE DOES NOT EXIST
C316  07714 002543R         DEF SYE11+1     WRONG NUMBER OF CALL PARAMETERS
C317  07715 000427R         DEF SYE12       NO CONSTANT WHERE EXPECTED
C318  07716 002565R         DEF SYE13       NO VARIABLE WHERE EXPECTED
C319  07717 002617R         DEF SYE14       NO CLOSING QUOTE FOR STRING
C320  07720 002631R         DEF SYE15       PRINT JUXTAPOSES FORMULAS
C321  07721 002675R         DEF SYE17       IMPROPER WORD IN MAT STATEMENT
C322  07722 002706R         DEF SYE18       NO COMMA WHERE EXPECTED
C323  07723 002761R         DEF SYE19       IMPROPER ARRAY FUNCTION
C324  07724 003001R         DEF SYE20       NO SUBSCRIPT WHERE EXPECTED
C325  07725 003013R         DEF SYE21       ARRAY INVERSION INTO SELF
C326  07726 003024R         DEF SYE22       MISSING MULTIPLICATION OPERATOR
C327  07727 003045R         DEF SYE23       IMPROPER ARRAY OPERATOR
C328  07730 003064R         DEF SYE24+1     ARRAY MULTIPLICATION INTO SELF
C329  07731 003206R         DEF FSCE1+1     MISSING LEFT PARENTHESIS
C330  07732 003262R         DEF FSCE2+1     MISSING RIGHT PARENTHESIS
C331  07733 003312R         DEF FSCE3+1     UNRECOGNIZED OPERAND
```

```
C352  07734 003540R        DEF ARRE1      MISSING SUBSCRIPT
C353  07735 003553R        DEF ARRE2      MISSING ARRAY IDENTIFIER
C354  07736 004374R        DEF SYE25+1    MISSING OR BAD INTEGER
C355  07737 000072R        DEF NOEOF+1    CHARACTERS AFTER STATEMENT END
C356  07740 003406R        DEF FSCE4+1    OUT OF CORE DURING SYNTAX
C357  07741 001754R        DEF PRERR      PHOTO READER NOT READY
C358  07742 005454R        DEF MER4       FUNCTION MULTIPLY DEFINED
C359  07743 005565R        DEF MER6       UNMATCHED FOR STATEMENT
C360  07744 005423R        DEF MER3       UNMATCHED NEXT
C361  07745 005671R        DEF MER8       OUT OF STORAGE-SYMBOL TABLE
C362  07746 005720R        DEF MSYM       INCONSISTENT DIMENSIONS
C363  07747 005442R        DEF MLOP6      LAST STATEMENT IS NOT 'END'
C364  07750 005551R        DEF MER5       ARRAY DOUBLE DIMENSIONED
C365  07751 005603R        DEF MER10      NO OF DIMENSIONS UNSPECIFIED
C366  07752 001144R        DEF MER9       ARRAY TOO LARGE
C367  07753 005640R        DEF MER7       OUT OF STORAGE-ARRAY ALLOCATION
C368  07754 007222R        DEF E6         SUBSCRIPT TOO LARGE
C369  07755 001303R        DEF E8         UNDEFINED OPERAND ACCESSED
C370  07756 007400R        DEF BASER      NEGATIVE BASE POWERED TO REAL
C371  07757 007505R        DEF POWER      ZERO TO ZERO POWER
C372  07760 005246R        DEF XEC5       MISSING STATEMENT
C373  07761 005570R        DEF E2         GOSUBS NESTED 10 DEEP
C374  07762 005574R        DEF E3         RETURN FINDS NO ADDRESS
C375  07763 005533R        DEF E4         OUT OF DATA
C376  07764 001241R        DEF E1+1       OUT OF STORAGE - EXECUTION
C377  07765 012000R        DEF E7         RE-DIMENSIONED ARRAY TOO LARGE
C378  07766 012067R        DEF LERR+1     DIMENSIONS NOT COMPATIBLE
C379  07767 012131R        DEF LCHK5      MATRIX UNASSIGNED
C380  07770 012670R        DEF LDUM1      NEARLY SINGULAR MATRIX
C381  07771 010425R        DEF TRGER      ARGUMENT TOO LARGE
C382  07772 010776R        DEF SQRER      SQRT HAS NEGATIVE ARGUMENT
C383  07773 011104R        DEF LOGER      LOG OF NEGATIVE ARGUMENT
C384  07774          RCERR EQU *          ** RECOVERABLE ERRORS FOLLOW  **
C385  07774 000727R        DEF OVRER      OVERFLOW
C386  07775 000723R        DEF UNDER      UNDERFLOW
C387  07776 011156R        DEF LNZR       LOG OF ZERO
C388  07777 011312R        DEF EXPER      EXPONENTIAL OVERFLOW
C389  10000 000000         NOP
C390  10001 007510R        DEF ZRTNG      ZERO TO NEGATIVE POWER
C391****** RECOVERABLE ERROR END *******
C392  10002 007003R        DEF WRERR
C393  10003 001546R        DEF NOCOR       OUT OF CORE DURING BUFFER ALLOCATION
C394  10004 015075R        DEF DIA1       INVALID DISK ADDR
C395  10005 002435R        DEF SYE1       INVALID COND. EXECUTABLE STATEMENT
C396  10006 002042R        DEF NOFIL       FILE NOT ON DISK
C397*
C398*
C399  10007 006412  EBUFF  OCT 6412
C400  10010 042522        ASC 3,ERROR
      10011 051117
      10012 051040
C401  10013 000000  EBFF   BSS 2
C402  10015 020111  LBUFF  ASC 5, IN LINE
      10016 047040
      10017 046111
```

```
        10020  047105
        10021  020040
0383    10022  000000  LNBFF BSS 3
```

```
C385**
C386***  PREDEFINED FUNCTION JUMP TABLE  **
C387**
C388   10025 007036R PDFT  DEF ETAB
C389   10026 010654R       DEF ESIN
C390   10027 010652R       DEF ECOS
C391   10030 010413R       DEF ETAN
C392   10031 010537R       DEF EATN
C393   10032 011204R       DEF EEXP
C394   10033 011075R       DEF ELOG
C395   10034 010647R       DEF EABS
C396   10035 010772R       DEF ESQR
C397   10036 011062R       DEF EINT
C398   10037 010743R       DEF ERND
C399   10040 011173R       DEF ESGN
C400**
C401***  OUTPUT A NUMBER  **
C402**
C403   10041 000000  NUMOT NOP             NUMBER IN (A) AND (B)
C404   10042 073366R       STA EXPON       SAVE NUMBER
C405   10043 002041        SEZ,RSS         SIGN?
C406   10044 026056R       JMP NS2         NO
C407   10045 002021        SSA,RSS         YES, NEGATIVE NUMBER?
C408   10046 026053R       JMP NS1         NO
C409   10047 017217R       JSB ARINV       YES, INVERT IT
C410   10050 073366R       STA EXPON
C411   10051 062150R       LDA .45
C412   10052 002001        RSS
C413   10053 062141R NS1   LDA .32         STORE
C414   10054 072041R       STA SIGN            SIGN
C415   10055 063366R       LDA EXPON
C416   10056 075351R NS2   STB GETDG       SAVE NUMBER
C417   10057 017160R       JSB IFIX        INTEGERIZE
C418   10060 000000        NOP
C419   10061 102201        SOC             WAS IT AN INTEGER?
C420   10062 026113R       JMP NUMO2       NO
C421**
C422***  OUTPUT AN INTEGER  **
C423**
C424   10063 002400        CLA             SAVE
C425   10064 075062R       STB B1+1          INTEGER
C426   10065 046255R       ADB M1000       5 OR MORE
C427   10066 005021        SSB,RSS           CHARACTERS?
C428   10067 040056        ADA .5          YES
C429   10070 040061        ADA .6          COMPUTE
C430   10071 042020R       ADA CCNT          END-OF-FIELD
C431   10072 003004        CMA,INA         SAVE MARKER TO
C432   10073 072060R       STA MLBX1+1       END-OF-FIELD
C433   10074 065272R       LDB PARM
C434   10075 005021        SSB,RSS
C435   10076 026102R       JMP *+4
C436   10077 042132R       ADA .21
C437   10100 002020        SSA
C438   10101 027002R       JMP WRERR-1
C439   10102 042164R       ADA .74         ENOUGH
```

```
C440   10103 002020          SSA                 ROOM?
C441   10104 017461R         JSB OUTLN
C442   10105 062041R         LDA SIGN
C443   10106 002002          SZA               SIGN?
C444   10107 017477R         JSB OUTCR         YES, OUTPUT IT
C445   10110 062062R         LDA B1+1          OUTPUT
C446   10111 017132R         JSB OUTIN               THE INTEGER
C447   10112 126041R         JMP NUMOT,I
C448   10113 003400  NUMO2   CCA               SET 'FIXED'
C449   10114 073037R         STA FFLAG           FLAG FALSE
C450   10115 063366R         LDA EXPON         LOAD
C451   10116 066351R         LDB GETDG           NUMBER
C452   10117 016002X         FAD MAXFX            IS NUMBER LESS THAN
       10120 000265R
C453   10121 002021          SSA,RSS               999999.5 ?
C454   10122 026132R         JMP NUMO5         NO
C455   10123 063366R         LDA EXPON         YES, IS
C456   10124 066351R         LDB GETDG           NUMBER
C457   10125 016002X         FAD MINFX           LESS THAN
       10126 000267R
C458   10127 066127R         LDB .12                 = 0.09999995?
C459   10130 002021          SSA,RSS
C460   10131 037037R         ISZ FFLAG         NO, SET FFLAG = 0 AND SKIP
C461   10132 064070  NUMO5   LDB .15           YES, LOAD 'FLOATING' FIELD WIDTH
C462   10133 046020R         ADB CCNT          SAVE
C463   10134 007004          CMB,INB             END-OF-FIELD
C464   10135 075060R         STB MLBX1+1           MARKER
C465   10136 045165R         ADB .75           ROOM
C466   10137 006020          SSB                 ENOUGH?
C467   10140 017461R         JSB OUTLN
C468**
C469***  OUTPUT A FLOATING POINT NUMBER  **
C470**
C471   10141 063366R         LDA EXPON
C472   10142 073066R         STA MANT1
C473   10143 066351R         LDB GETDG         UNPACK
C474   10144 017223R         JSB .FLUN
C475   10145 077130R         STB MANT2           NUMBER
C476   10146 072042R         STA EXP
C477   10147 062041R         LDA SIGN
C478   10150 002002          SZA               SIGN?
C479   10151 017477R         JSB OUTCR         YES, OUTPUT IT
C480   10152 002400          CLA               INITIALIZE COUNTER
C481   10153 073366R         STA EXPON           FOR DECIMAL EXPONENT
C482   10154 052042R         CPA EXP           EXPONENT ZERO?
C483   10155 026200R         JMP EOUT4         YES
C484   10156 016777R EOUT2   JSB MBY10         NO,
C485   10157 062042R         LDA EXP             MULTIPLY
C486   10160 003004          CMA,INA               NUMBER BY 10
C487   10161 002020          SSA                     UNTIL
C488   10162 026165R         JMP *+3                   IT IS
C489   10163 037366R         ISZ EXPON                   GREATER
C490   10164 026156R         JMP EOUT2                     THAN 1
C491   10165 017030R         JSB DBY10         DIVIDE BY 10
C492   10166 063366R         LDA EXPON
```

```
C493  10167 066042R EOUT3 LDB EXP       DIVIDE
C494  10170 007004        CMB,INB       NUMBER
C495  10171 006021        SSB,RSS         BY 10
C496  10172 026200R       JMP EOUT4         UNTIL
C497  10173 073366R       STA EXPON           IT IS
C498  10174 017030R       JSB OBY10             LESS
C499  10175 003400        CCA                     THAN
C500  10176 043366R       ADA EXPON                 1
C501  10177 025167R       JMP EOUT3
C502  10200 003000  EOUT4 CMA           SET EXPONENT
C503  10201 073366R       STA EXPON     TO TRUE VALUE-1
C504  10202 064044        LDB M7        SET DIGIT
C505  10203 076375R       STB RETCR     COUNTER
C506  10204 007400        CCB           SET DECIMAL
C507  10205 077461R       STB OUTLN     POINT FLAG
C508  10206 057037R       CPB FFLAG     FIXED POINT?
C509  10207 026216R       JMP EOUT6     NO
C510  10210 003000        CMA           YES, SET
C511  10211 073461R       STA OUTLN     DECIMAL POINT FLAG
C512  10212 050054        CPA .1        .1?
C513  10213 026222R       JMP EOUT5     YES
C514  10214 002021        SSA,RSS       LEADING DECIMAL POINT?
C515  10215 026230R       JMP EOUT7+2   YES
C516  10216 016351R EOUT6 JSB GETDG     OUTPUT
C517  10217 042154R       ADA .48         A
C518  10220 017477R       JSB OUTCR         DIGIT
C519  10221 026232R       JMP EOUT8
C520  10222 062152R EOUT5 LDA .46       OUTPUT
C521  10223 017477R       JSB OUTCR     DECIMAL POINT
C522  10224 062154R       LDA .48       OUTPUT
C523  10225 026231R       JMP EOUT8-1   LEADING ZERO
C524  10226 057461R EOUT7 ISZ OUTLN     DECIMAL POINT NEXT?
C525  10227 026216R       JMP EOUT6     NO
C526  10230 062152R       LDA .46       YES,
C527  10231 017477R       JSB OUTCR     OUTPUT IT
C528  10232 056375R EOUT8 ISZ RETCR     MORE MANTISSA?
C529  10233 026226R       JMP EOUT7     YES
C530  10234 062020R       LDA CCNT      NO,
C531  10235 072347R       STA NUMO1     SAVE
C532  10236 062017R       LDA BADDR       OUTPUT
C533  10237 072350R       STA NUMO3         POINTERS
C534  10240 016351R       JSB GETDG     NEXT DIGIT
C535  10241 040046        ADA M5        FIVE OR
C536  10242 002020        SSA             GREATER?
C537  10243 026506R       JMP EOUT1     NO
C538  10244 003400        CCA           SET DECIMAL
C539  10245 073066R ERND1 STA SYMCK     POINT COUNTER
C540  10246 016375R       JSB RETCR     RETRIEVE CHARACTER
C541  10247 052152R       CPA .46       DECIMAL POINT?
C542  10250 026244R       JMP ERND1-1   YES, FLAG IT
C543  10251 017353R       JSB DIGCK     NO, DIGIT?
C544  10252 026265R       JMP ERND2     NO
C545  10253 050064        CPA .9        YES, 9?
C546  10254 026257R       JMP *+3       YES
C547  10255 042155R       ADA .49       NO, BUMP
```

```
C548  10256 026501R        JMP ERND3       DIGIT 1
C549  10257 062154R        LDA .48    OVERLAY
C550  10260 017477R        JSB OUTCR       A ZERO
C551  10261 016375R        JSB RETCR     BACKSPACE
C552  10262 003400         CCA           DECREMENT
C553  10263 043066R        ADA SYMCK       DECIMAL POINT
C554  10264 026245R        JMP ERND1         COUNTER
C555  10265 017477R ERND2  JSB OUTCR     RESTORE CHARACTER
C556  10266 057366R        ISZ EXPON     CORRECT
C557  10267 000000         NOP             EXPONENT
C558  10270 062155R        LDA .49       OVERLAY A 1
C559  10271 067037R        LDB FFLAG     FIXED
C560  10272 006002         SZB             POINT?
C561  10273 026301R        JMP ERND3     NO
C562  10274 017477R        JSB OUTCR     YES, OUTPUT CHARACTER
C563  10275 062154R        LDA .48       PREPARE TO OVERLAY A ZERO
C564  10276 057066R        ISZ SYMCK     DECIMAL POINT NEXT?
C565  10277 026274R        JMP *-3       NO
C566  10300 062152R        LDA .46       YES
C567  10301 017477R ERND3  JSB OUTCR
C568  10302 062347R        LDA NUMO1     RESTORE
C569  10303 072020R        STA CCNT        OUTPUT
C570  10304 062350R        LDA NUMO3         POINTERS
C571  10305 072017R        STA BADDR
C572  10306 057037R EOUT1  ISZ FFLAG      NO, FIXED POINT?
C573  10307 026336R        JMP EOUT9     YES
C574  10310 062161R        LDA E         NO,
C575  10311 017477R        JSB OUTCR       OUTPUT 'E'
C576  10312 062150R        LDA .45       LOAD '-'
C577  10313 067366R        LDB EXPON     POSITIVE
C578  10314 006020         SSB             EXPONENT?
C579  10315 007005         CMB,INB,RSS   NO
C580  10316 062147R        LDA .43       YES, LOAD '+'
C581  10317 077366R        STB EXPON
C582  10320 017477R        JSB OUTCR     OUTPUT SIGN
C583  10321 067366R        LDB EXPON
C584  10322 062154R        LDA .48       COMPUTE
C585  10323 044041         ADB M10
C586  10324 006020         SSB             EXPONENT
C587  10325 026330R        JMP *+3
C588  10326 002004         INA               DIGIT
C589  10327 026323R        JMP *-4
C590  10330 046157R        ADB .58       COMPUTE
C591  10331 077366R        STB EXPON       SECOND DIGIT
C592  10332 017477R        JSB OUTCR     OUTPUT
C593  10333 063366R        LDA EXPON
C594  10334 017477R        JSB OUTCR       EXPONENT
C595  10335 126041R        JMP NUMOT,I
C596  10336 016375R EOUT9  JSB RETCR     RETRIEVE CHARACTER
C597  10337 052154R        CPA .48       ZERO?
C598  10340 026343R        JMP EOUT0     YES
C599  10341 017477R        JSB OUTCR     NO, RESTORE CHARACTER
C600  10342 126041R        JMP NUMOT,I
C601  10343 062141R EOUT0  LDA .32       OVERLAY
C602  10344 017477R        JSB OUTCR       A BLANK
```

```
C603  10345 016575R         JSB RETCR      BACKSPACE
C604  10346 026336R         JMP EOUT9
C605  10347 000000   NUM01  BSS 1
C606  10350 000000   NUM03  BSS 1
```

```
C608**
C609***  GET DIGIT TO OUTPUT  **
C610**
C611  10351 000000  GETDG NOP
C612  10352 016777R       JSB MBY10      MULTIPLY BY 10
C613  10353 066042R       LDB EXP        GET EXPONENT IN (B)
C614  10354 007004        CMB,INB          AS NEGATIVE
C615  10355 012230R       AND HIMSK      KEEP 5 HIGH BITS OF (A)
C616  10356 001200        RAL            NORMALIZE TO BIT 15
C617  10357 006024        SSB,INB        ROTATE INTEGER
C618  10360 026356R       JMP *-2          INTO (A)
C619  10361 010074        AND MSK0       EXTRACT
C620  10362 072430R       STA NUMCK        DIGIT
C621  10363 066042R       LDB EXP        ROTATE
C622  10364 007004        CMB,INB
C623  10365 001300        RAR              BACK
C624  10366 006024        SSB,INB
C625  10367 026365R       JMP *-2
C626  10370 023066R       XOR MANT1      REMOVE
C627  10371 067130R       LDB MANT2        DIGIT
C628  10372 016742R       JSB NORML      NORMALIZE REMAINDER
C629  10373 062430R       LDA NUMCK      LOAD (A) WITH DIGIT
C630  10374 126351R       JMP GETDG,I
C631**
C632***  RETRIEVE CHARACTER FROM OUTPUT BUFFER  **
C633**
C634  10375 000000  RETCR NOP
C635  10376 066020R       LDB CCNT       DECREMENT
C636  10377 044052        ADB M1           CHARACTER
C637  10400 076020R       STB CCNT           COUNT
C638  10401 162017R       LDA BADDR,I    POSITION
C639  10402 006011        SLB,RSS          AND
C640  10403 001727        ALF,ALF            EXTRACT
C641  10404 010074        AND MSK0             CHARACTER
C642  10405 004010        SLB            FIRST CHARACTER OF WORD?
C643  10406 126375R       JMP RETCR,I    NO
C644  10407 066017R       LDB BADDR      YES, DECREMENT
C645  10410 044052        ADB M1           BUFFER
C646  10411 076017R       STB BADDR          POINTER
C647  10412 126375R       JMP RETCR,I
C648*
C649*
```

```
C651*                ******************************
C652*                SUBROUTINE TO CALCULATE TAN(X)
C653*                ******************************
C654*
C655*                CALLED BY A JMP ETAN WITH THE ARGUMENT
C656*                IN FLOATING RADIANS IN THE REGISTERS.
C657*                THE FLOATING RESULT IS RETURNED IN A & B
C658*
C659   10413 016001X ETAN   FMP FOPI
       10414 010502R
C660   10415 072506R        STA XTEMP
C661   10416 076507R        STB XTEMP+1
C662   10417 016002X        FAD K1
       10420 010504R
C663   10421 017445R        JSB .PWR2
C664   10422 177776         DEC -2
C665   10423 017420R        JSB .IENT
C666   10424 016274R        JSB ERROR
C667   10425 017437R TRGER  JSB FLOAT
C668   10426 017217R        JSB ARINV
C669   10427 017445R        JSB .PWR2
C670   10430 000002         DEC 2
C671   10431 016002X        FAD XTEMP
       10432 010506R
C672   10433 072506R        STA XTEMP
C673   10434 076507R        STB XTEMP+1    X=X-4*ENTIER((X+1)/4)
C674   10435 016003X        FSB K1
       10436 010504R
C675   10437 073037R        STA SBOX       TEMPORARY
C676   10440 002020         SSA            X<1?
C677   10441 025474R        JMP ELSE1      YES
C678   10442 062514R        LDA K2         NO
C679   10443 066515R        LDB K2+1
C680   10444 016003X        FSB XTEMP
       10445 010506R
C681   10446 072510R BOTH1  STA YTEMP
C682   10447 076511R        STB YTEMP+1    Y= 2-X
C683   10450 016001X        FMP YTEMP
       10451 010510R
C684   10452 016001X        FMP K2
       10453 010514R
C685   10454 016003X        FSB K1
       10455 010504R
C686   10456 017331R        JSB .CHEB
C687   10457 010516R        DEF COEFF
C688   10460 016001X        FMP YTEMP
       10461 010510R
C689   10462 072510R        STA YTEMP
C690   10463 076511R        STB YTEMP+1    Y=Y*CHEBY(2*Y**2-1)
C691   10464 063037R        LDA SBOX
C692   10465 002020         SSA            X<1 ?
C693   10466 025477R        JMP ELSE2      YES
C694   10467 062504R        LDA K1
C695   10470 066505R        LDB K1+1
C696   10471 016004X        FDV YTEMP
```

```
      10472 010510R
C697  10473 025101R         JMP FOR12
C698  10474 062506R ELSE1   LDA XTEMP
C699  10475 066507R         LDB XTEMP+1
C700  10476 026446R         JMP BOTH1      Y=X
C701  10477 062510R ELSE2   LDA YTEMP
C702  10500 066511R         LDB YTEMP+1
C703  10501 025101R         JMP FOR12
C704*
C705  10502 050574  FOP1    DEC 1.273239545     4/PI
      10503 141092
C706  10504 040000  K1      DEC 1.
      10505 000002
C707  10506 000000  XTEMP   BSS 2
C708  10510 000000  YTEMP   BSS 2
C709  10512 000000  UTEMP   BSS 2
C710  10514 040000  K2      DEC 2.
      10515 000004
C711  10516 076061  COEFF   DEC 1.4458E-8
      10517 075315
C712  10520 066034          DEC 2.013766E-7
      10521 176725
C713  10522 057035          DEC 2.804816E-6
      10523 030335
C714  10524 050755          DEC 5.906637E-5
      10525 114745
C715  10526 043523          DEC 5.4417038E-4
      10527 052355
C716  10530 076112          DEC 7.586101578E-3
      10531 065763
C717  10532 066520          DEC .10675392857
      10533 163773
C718  10534 070512          DEC 1.7701474227
      10535 014002
C719  10536 000000          OCT 0
```

```
C721*                ********************************
C722*                SUBROUTINE TO CALCULATE ATN(X)
C723*                ********************************
C724*
C725*                CALLED BY A JMP EATN WITH THE ARGUMENT
C726*                IN FLOATING POINT FORM IN THE REGISTERS.
C727*                THE FLOATING RESULT IN THE RANGE -PI/2
C728*                TO PI/2 IS RETURNED IN A & B
C729*
C730   10537 072506R EATN  STA XTEMP
C731   10540 076507R       STB XTEMP+1
C732   10541 060001        LDA 1
C733   10542 010074        AND MSK0
C734   10543 073037R       STA SBOX        TAN = EXP OF (X)
C735   10544 002002        SZA
C736   10545 000010        SLA             ABS (X) > 1 ?
C737   10546 026603R       JMP ELS1        NO
C738   10547 062504R       LDA K1
C739   10550 066505R       LDB K1+1
C740   10551 016004X       FDV XTEMP       U=1/X
       10552 010506R
C741   10553 072512R BTH1  STA UTEMP
C742   10554 076513R       STB UTEMP+1
C743   10555 016001X       FMP UTEMP
       10556 010512R
C744   10557 016001X       FMP K2
       10560 010514R
C745   10561 016003X       FSB K1
       10562 010504R
C746   10563 017331R       JSB .CHEB
C747   10564 010622R       DEF COEF
C748   10565 016001X       FMP UTEMP
       10566 010512R
C749   10567 072510R       STA YTEMP
C750   10570 076511R       STB YTEMP+1     Y=U*CHEBY(2*U**2-1)
C751   10571 063037R       LDA SBOX
C752   10572 002002        SZA
C753   10573 000010        SLA             ABS(X)>1 ?
C754   10574 026606R       JMP ELS2        NO
C755   10575 062506R       LDA XTEMP
C756   10576 002020        SSA             X<0 ?
C757   10577 026611R       JMP ELS3        YES
C758   10600 062616R       LDA PIBY2
C001   10601 066617R       LDB PIBY2+1
C002   10602 026613R       JMP ELS3+2      ANS = PI/2 - Y
C003   10603 062506R ELS1  LDA XTEMP
C004   10604 066507R       LDB XTEMP+1
C005   10605 026553R       JMP BTH1        U=X
C006   10606 062510R ELS2  LDA YTEMP
C007   10607 066511R       LDB YTEMP+1
C008   10610 026101R       JMP FOR12        ANSWER = Y
C009   10611 062620R ELS3  LDA MP2
C010   10612 066621R       LDB MP2+1
C011   10613 016003X       FSB YTEMP
       10614 010510R
```

```
C012  10615 025101R        JMP FOR12
C013*
C014  10616 062207  PI8Y2  DEC 1.5707963268   PI/2
      10617 165402
C015  10620 115570  MP2    DEC -1.5707963268  -PI/2
      10621 011402
C016  10622 105671  COEF   DEC -1.53034E-8
      10623 102315
C017  10624 056335         DEC 8.64888E-8
      10625 156323
C018  10626 131601         DEC -56.99186E-8
      10627 157731
C019  10630 040033         DEC 3.821037E-6
      10631 035737
C020  10632 111013         DEC -2.6215196E-5
      10633 123345
C021  10634 060342         DEC 1.8574297E-4
      10635 000351
C022  10636 122573         DEC -1.381195004E-3
      10637 062757
C023  10640 055471         DEC .01113584206
      10641 107365
C024  10642 111620         DEC -.1058929245
      10643 147373
C025  10644 070320         DEC 1.762747174
      10645 135002
C026  10646 000000         OCT 0
```

```
C028*              ********************************
C029*              SUBROUTINE TO COMPUTE ABS(X)
C030*              ********************************
C031*
C032*              CALLED BY A JMP EABS WITH THE ARGUMENT
C033*              IN FLOATING POINT FORM IN THE REGISTERS.
C034*              THE ABSOLUTE VALUE IN FLOATING POINT
C035*              IS RETURNED IN A & B
C036*
C037  10647 002020  EABS  SSA            NEGATIVE?
C038  10650 017217R       JSB ARINV      YES, NEGATE IT
C039  10651 025101R       JMP FOR12
```

```
C041*                ******************************
C042*                SUBROUTINE TO CALCULATE SIN(X)
C043*                ******************************
C044*
C045*                CALLED BY A JMP ESIN WITH THE ARGUMENT
C046*                IN FLOATING RADIANS IN THE REGISTERS.
C047*                THE FLOATING RESULT IS RETURNED IN A & B
C048*
C049   10652 016002X ECOS  FAD PIBY2
       10653 010616R
C050   10654 016001X ESIN  FMP TOPI
       10655 010724R
C051   10656 072506R       STA XTEMP
C052   10657 076507R       STB XTEMP+1    X=2*X/PI
C053   10660 016002X       FAD K1
       10661 010504R
C054   10662 017445R       JSB .PWR2
C055   10663 177776        DEC -2
C056   10664 017420R       JSB .IENT
C057   10665 026424R       JMP IRGER-1    ERROR IF EXPONENT >= 15
C058   10666 017437R       JSB FLOAT
C059   10667 016001X       FMP MM4
       10670 010726R
C060   10671 016002X       FAD XTEMP
       10672 010506R
C061   10673 072506R       STA XTEMP
C062   10674 076507R       STB XTEMP+1    X=X-4*ENTIER((X+1)/4)
C063   10675 016003X       FSB K1
       10676 010504R
C064   10677 002020        SSA            X<1 ?
C065   10700 025707R       JMP PAST       YES
C066   10701 062514R       LDA K2
C067   10702 066515R       LDB K2+1
C068   10703 016003X       FSB XTEMP
       10704 010506R
C069   10705 072506R       STA XTEMP
C070   10706 076507R       STB XTEMP+1    X=2-X
C071   10707 062506R PAST  LDA XTEMP
C072   10710 066507R       LDB XTEMP+1
C073   10711 016001X       FMP XTEMP
       10712 010506R
C074   10713 017445R       JSB .PWR2
C075   10714 000001        DEC 1
C076   10715 016003X       FSB K1
       10716 010504R
C077   10717 017331R       JSB .CHEB
C078   10720 010730R       DEF COEF1
C079   10721 016001X       FMP XTEMP
       10722 010506R
C080   10723 025101R       JMP FOR12      ANS = X+CHEBY (2*X**2-1)
C081*
C082   10724 050574  TOPI  DEC .636619772      2/PI
       10725 141000
C083   10726 100000  MM4   DEC -4.
       10727 000004
```

```
C084  10730 047605  COEF1 DEC 1.18496E-6
      10731 072733
C085  10732 134143        DEC -1.365875E-4
      10733 104751
C086  10734 045261        DEC 9.118016E-3
      10735 157365
C087  10736 133571        DEC -.2852615692
      10737 014777
C088  10740 050656        DEC 2.5525579248
      10741 107004
C089  10742 000000        OCT 0




C091*             ******************************
C092*             SUBROUTINE TO COMPUTE RND(X)
C093*             ******************************
C094*
C095*             THE ARGUMENT OF RND IS A DUMMY ONE
C096*             THE ROUTINE GENERATES A RANDOM NUMBER
C097*             IN THE A & B REGISTERS
C098*
C099*  R=X/M, X=C*X MOD M, M=2^31, C=2^15 + 3
C100*
C101  10743 002400  ERND  CLA            X IS INITIALLY 1
C102  10744 072042R       STA EXP        INITIALIZE EXPONENT
C103  10745 062043R       LDA XH         COMPUTE
C104  10746 001000       ALS              HIGH
C105  10747 042043R       ADA XH             PART
C106  10750 066044R       LDB XL               2*XH
C107  10751 004065       CLE,ERB                + XH +
C108  10752 040001       ADA 1                    XL*2^15
C109  10753 066044R       LDB XL
C110  10754 005275       RBL,CLE,SLB,ERB     ADD XL[15] TO
C111  10755 002004       INA                   (A) (FROM 2*XL)
C112  10756 004066       CLE,ELB             2*XL
C113  10757 046044R       ADB XL                + XL
C114  10760 001675       ELA,CLE,SLA,ERA     ADD OVERFLOW
C115  10761 002104       CLE,INA               TO (A)
C116  10762 046263R       ADB FLGBT      ADD IN TRAILING BIT OF XL*2^15
C117  10763 002040       SEZ            ADD OVERFLOW
C118  10764 002004       INA              TO (A)
C119  10765 001665       ELA,CLE,ERA    ERASE A[15]
C120  10766 072043R       STA XH         STORE
C121  10767 076044R       STB XL           INTEGER
C122  10770 016647R       JSB .PACK      NORMALIZE AND PACK
C123  10771 026101R       JMP FOR12
```

```
C125*                ********************************
C126*                SUBROUTINE TO CALCULATE SQR(X)
C127*                ********************************
C128*
C129*                CALLED BY A JMP ESQR WITH THE ARGUMENT
C130*                IN FLOATING POINT FORM IN THE REGISTERS.
C131*                THE FLOATING RESULT IS RETURNED IN A & B
C132*
C133  10772 002003  ESQR  SZA,RSS        X=0 ?
C134  10773 026101R       JMP FOR12      YES, ANS = 0
C135  10774 002020       SSA            X<0 ?
C136  10775 016274R       JSB ERROR      YES ERROR
C137  10776 072506R SQRER STA XTEMP
C138  10777 017225R       JSB .FLUN
C139  11000 000031       SLA,ARS        EXP(X) ODD?
C140  11001 027041R       JMP ODD
C141  11002 040052       ADA M1
C142  11003 073037R       STA SBOX       SBOX=EXPO(X)/2-1
C143  11004 076507R       STB XTEMP+1    LOW MANTISSA/2
C144  11005 062506R       LDA XTEMP
C145  11006 016001X       FMP SA2
      11007 011054R
C146  11010 016002X       FAD SB2
      11011 011060R
C147  11012 072510R BTH2  STA YTEMP
C148  11013 076511R       STB YTEMP+1
C149  11014 062506R       LDA XTEMP
C150  11015 066507R       LDB XTEMP+1
C151  11016 016004X       FDV YTEMP
      11017 010510R
C152  11020 016002X       FAD YTEMP
      11021 010510R
C153  11022 017445R       JSB .PWR2
C154  11023 177777       DEC -1
C155  11024 072510R       STA YTEMP
C156  11025 076511R       STB YTEMP+1    Y=(Y+X/Y)/2
C157  11026 062506R       LDA XTEMP
C158  11027 066507R       LDB XTEMP+1
C159  11030 016004X       FDV YTEMP
      11031 010510R
C160  11032 016002X       FAD YTEMP
      11033 010510R
C161  11034 072510R       STA YTEMP
C162  11035 076511R       STB YTEMP+1
C163  11036 017445R       JSB .PWR2
C164  11037 000000  SBOX  OCT 0
C165  11040 026101R       JMP FOR12      ANS = (P+F/P)*2**SBOX
C166  11041 073037R ODD   STA SBOX       SBOX = EXP(X)/2
C167  11042 044074       ADB MSK0
C168  11043 076507R       STB XTEMP+1    LOW MANTISSA/2
C169  11044 062506R       LDA XTEMP
C170  11045 016001X       FMP SA1
      11046 011052R
C171  11047 016002X       FAD SB1
      11050 011056R
```

```
C172  11051 027012R        JMP BTH2       Y=SB1+SA1*X
C173*
C174  11052 070000  SA1    DEC .875
      11053 000000
C175  11054 045000  SA2    DEC .578125
      11055 000000
C176  11056 043524  SB1    DEC .27863
      11057 046377
C177  11060 066000  SB2    DEC .421875
      11061 000377



C179*               ********************************
C180*               SUBROUTINE TO CALCULATE INT(X)
C181*               ********************************
C182*
C183*               CALLED BY A JMP EINT WITH THE ARGUMENT
C184*               IN FLOATING POINT FORM IN THE REGISTERS.
C185*               ENTIER(X) IS FLOATED AND RETURNED IN A & B
C186*
C187*               CALCULATES THE FLOATING POINT EQUIVALENT
C188*               OF ENTIER(X)
C189*
C190  11062 077037R EINT   STB SBOX       SAVE LOW WORD
C191  11063 064071        LDB .31        PRESET BIAS FOR
C192  11064 076042R        STB EXP          BINARY POINT
C193  11065 067037R        LDB SBOX       24-BIT
C194  11066 017160R        JSB IFIX         INTEGER?
C195  11067 027072R        JMP EINT1      NO
C196  11070 015647R        JSB .PACK      YES, PACK IT
C197  11071 026101R        JMP FOR12
C198  11072 063377R EINT1  LDA GETCR      RETRIEVE
C199  11073 067037R        LDB SBOX         NUMBER
C200  11074 026101R        JMP FOR12
```

```
C202*                ********************************
C203*                SUBROUTINE TO CALCULATE LOG(X)
C204*                ********************************
C205*
C206*                CALLED BY A JMP ELOG WITH THE ARGUMENT
C207*                IN FLOATING POINT FORM IN THE REGISTERS.
C208*                THE FLOATING RESULT IS RETURNED IN A & B
C209*
C210   11075 017077R ELOG  JSB .LOG
C211   11076 026101R       JMP FOR12
C212   11077 000000  .LOG  NOP
C213   11100 002003        SZA,RSS          NON-ZERO ARGUMENT?
C214   11101 027155R       JMP .LOG1        NO
C215   11102 002020        SSA              YES, POSITIVE ARGUMENT?
C216   11103 015274R       JSB ERROR        NO
C217   11104 072506R LOGER STA XTEMP        YES
C218   11105 017223R       JSB .FLUN
C219   11106 076507R       STB XTEMP+1
C220   11107 017437R       JSB FLOAT
C221   11110 072510R       STA YTEMP
C222   11111 076511R       STB YTEMP+1      Y=EXPO(X)
C223   11112 062506R       LDA XTEMP
C224   11113 066507R       LDB XTEMP+1
C225   11114 016002X       FAD R22
       11115 011161R
C226   11116 072512R       STA UTEMP
C227   11117 076513R       STB UTEMP+1      U=X+SQR(0.5)
C228   11120 062506R       LDA XTEMP
C229   11121 066507R       LDB XTEMP+1
C230   11122 016003X       FSB R22
       11123 011161R
C231   11124 016004X       FDV UTEMP
       11125 010512R
C232   11126 072512R       STA UTEMP
C233   11127 076513R       STB UTEMP+1      U=(X-SQR(0.5))/U
C234   11130 016001X       FMP UTEMP
       11131 010512R
C235   11132 016003X       FSB CCC
       11133 011171R
C236   11134 072506R       STA XTEMP
C237   11135 076507R       STB XTEMP+1
C238   11136 063167R       LDA MB
C239   11137 067170R       LDB MB+1
C240   11140 016004X       FDV XTEMP
       11141 010506R
C241   11142 016002X       FAD AAA
       11143 011165R
C242   11144 016001X       FMP UTEMP
       11145 010512R
C243   11146 016003X       FSB HALF
       11147 009261R
C244   11150 016002X       FAD YTEMP
       11151 010510R
C245   11152 016001X       FMP LE2
       11153 011163R
```

```
C246  11154 127077R        JMP .LOG,I     ANS=LOG(2)*(EXPO(X)-0.5+U*
C247*                                      (A-B/X))
C248  11155 015274R .LOG1  JSB ERROR      LOG OF ZERO
C249  11156 062263R LNZR   LDA MNEG       RETURN
C250  11157 066175R        LDB B776         NEGATIVE
C251  11160 127077R        JMP .LOG,I         INFINITY
C252*
C253  11161 055202  R22    DEC .707106781      SQR(0.5)
      11162 075000
C254  11163 054271  LE2    DEC .6931471806     LOG BASE E OF 2
      11164 006000
C255  11165 051260  AAA    DEC 1.2920070987
      11166 057402
C256  11167 125606  MB     DEC -2.6398577035
      11170 044404
C257  11171 065010  CCC    DEC 1.6567626301
      11172 063002



C259*               ********************************
C260*               SUBROUTINE TO COMPUTE SGN(X)
C261*               ********************************
C262*
C263*               CALLED BY A JMP ESGN WITH THE ARGUMENT
C264*               IN FLOATING POINT FORM IN THE REGISTERS.
C265*               ON RETURN A & B CONTAIN THE FOLLOWING:
C266*
C267*               IF X>0 THEN +1.
C268*               IF X=0 THEN  0
C269*               IF X<0 THEN -1.
C270*
C271  11173 006400  ESGN   CLB
C272  11174 002003         SZA,RSS        ZERO?
C273  11175 026101R        JMP FOR12           YES
C274  11176 002021         SSA,RSS        NO, POSITIVE?
C275  11177 064055         LDB .2         YES, SET EXPONENT
C276  11200 062263R        LDA FLGBT      LOAD MANTISSA
C277  11201 006002         SZB            POSITIVE?
C278  11202 001300         RAR            YES, CORRECT MANTISSA
C279  11203 026101R        JMP FOR12
```

```
C281*              ********************************
C282*              SUBROUTINE TO CALCULATE EXP(X)
C283*              ********************************
C284*
C285*              CALLED BY A JMP EEXP WITH THE ARGUMENT
C286*              IN FLOATING POINT FORM IN THE REGISTERS.
C287*              THE FLOATING RESULT IS RETURNED IN A & B
C288*
C289  11204 017206R EEXP  JSB .EXP
C290  11205 026101R       JMP FOR12
C291  11206 000000  .EXP  NOP
C292  11207 016001X       FMP L2E
      11210 011527R
C293  11211 072506R       STA XTEMP
C294  11212 076507R       STB XTEMP+1    X=Z*LOG2(E)
C295  11213 017420R       JSB .IENT
C296  11214 027306R       JMP .EXP1
C297  11215 073501R       STA INTE       INTE = ENTIER(X)
C298  11216 017437R       JSB FLOAT
C299  11217 072510R       STA YTEMP
C300  11220 076511R       STB YTEMP+1    Y=ENTIER(X)
C301  11221 063301R       LDA INTE
C302  11222 043315R       ADA M124
C303  11223 002021        SSA,RSS        INTE >=124 ?
C304  11224 027511R       JMP EXPER-1    YES,ERROR
C305  11225 043316R       ADA .244       INTE <-120 ?
C306  11226 002020        SSA
C307  11227 027303R       JMP ZERE       YES,ANS=0
C308  11230 062506R       LDA XTEMP
C309  11231 066507R       LDB XTEMP+1
C310  11232 016003X       FSB YTEMP
      11233 010510R
C311  11234 072506R       STA XTEMP
C312  11235 076507R       STB XTEMP+1    X=X-ENTIER(X)
C313  11236 016001X       FMP XTEMP
      11237 010506R
C314  11240 072512R       STA UTEMP
C315  11241 076513R       STB UTEMP+1    U=X**2
C316  11242 016002X       FAD AAAA
      11243 011517R
C317  11244 072510R       STA YTEMP
C318  11245 076511R       STB YTEMP+1    Y=X**2+AAAA
C319  11246 063321R       LDA BBBB
C320  11247 067322R       LDB BBBB+1
C321  11250 016004X       FDV YTEMP
      11251 010510R
C322  11252 072510R       STA YTEMP
C323  11253 076511R       STB YTEMP+1    Y=BBBB/Y
C324  11254 063325R       LDA CCCC
C325  11255 067324R       LDB CCCC+1
C326  11256 016001X       FMP UTEMP
      11257 010512R
C327  11260 016002X       FAD DDDD
      11261 011325R
C328  11262 016003X       FSB XTEMP
```

```
        11263 010506R
C329    11264 016003X           FSB YTEMP
        11265 010510R
C330    11266 072510R           STA YTEMP
C331    11267 076511R           STB YTEMP+1     Y=-X+DDDD+CCCC*X**2-Y
C332    11270 062506R           LDA XTEMP
C333    11271 066507R           LDB XTEMP+1
C334    11272 016004X           FDV YTEMP
        11273 010510R
C335    11274 016002X           FAD HALF
        11275 000261R
C336    11276 037501R           ISZ INTE
C337    11277 000000            NOP
C338    11300 017445R           JSB .PWR2
C339    11301 000000  INTE      OCT 0
C340    11302 127206R           JMP .EXP,I      ANS=(0.5+X/Y)*2**INTE
C341    11303 002400  ZERE      CLA
C342    11304 006400            CLB
C343    11305 127206R           JMP .EXP,I      ANS=0
C344    11306 063405R .EXP1     LDA X2TMP
C345    11307 002020            SSA
C346    11310 027303R           JMP ZERE
C347    11311 016274R           JSB ERROR
C348    11312 062222R EXPER     LDA INF
C349    11313 064051            LDB M2
C350    11314 127206R           JMP .EXP,I
C351*
C352    11315 177604  M124      DEC -124
C353    11316 000364  .244      DEC 244
C354    11317 053552  AAAA      DEC 87.417497202
        11320 160416
C355    11321 046477  BBBB      DEC 617.9722695
        11322 016424
C356    11323 043576  CCCC      DEC .03465735903
        11324 070371
C357    11325 047643  DDDD      DEC 9.9545957821
        11326 001410
C358    11327 056125  L2E       DEC 1.4426950409
        11330 017002
```

```
C360*                *******************************
C361*                SUBROUTINE TO COMPUTE CHEBY(X)
C362*                *******************************
C363*
C364*                CALLING SEQUENCE:
C365*
C366*                LDA X           (FLOATING)
C367*                LDB X+1
C368*                JSB .CHEB       (RESULT FLOATING)
C369*                DEF C           (TABLE OF CHEBY.COEFFS.,FLOATING)
C370*
C371   11331 000000  .CHEB NOP
C372   11332 016001X       FMP K2
       11333 010514R
C373   11334 073406R       STA X2TMP
C374   11335 077407R       STB X2TMP+1    X2 =X*2
C375   11336 167331R       LDB .CHEB,I
C376   11337 077414R       STB CTMP       C POINTS TO COEFFICIENT TABLE
C377   11340 160001        LDA 1,I
C378   11341 006004        INB
C379   11342 164001        LDB 1,I        GET FIRST COEFF
C380   11343 073416R       STA DTMP
C381   11344 077417R       STB DTMP+1     D=C(N)
C382   11345 002400        CLA
C383   11346 073412R       STA BTMP
C384   11347 073413R       STA BTMP+1     B=0
C385   11350 037414R LOPC  ISZ CTMP
C386   11351 037414R       ISZ CTMP       N=N-1
C387   11352 163414R       LDA CTMP,I
C388   11353 002003        SZA,RSS        C(N)=0 ?
C389   11354 027376R       JMP COUT       ZERO FLAGS END OF TABLE
C390   11355 063412R       LDA BTMP       NO
C391   11356 067413R       LDB BTMP+1
C392   11357 073410R       STA ATMP
C393   11360 077411R       STB ATMP+1     A=B
C394   11361 063416R       LDA DTMP
C395   11362 067417R       LDB DTMP+1
C396   11363 073412R       STA BTMP
C397   11364 077413R       STB BTMP+1     B=D
C398   11365 016001X       FMP X2TMP
       11366 011406R
C399   11367 016003X       FSB ATMP
       11370 011410R
C400   11371 016002X       FAD CTMP,I
       11372 111414R
C401   11373 073416R       STA DTMP
C402   11374 077417R       STB DTMP+1     D=C(N) -A+B*X2
C403   11375 027350R       JMP LOPC
C404   11376 063416R COUT  LDA DTMP
C405   11377 067417R       LDB DTMP+1
C406   11400 016003X       FSB ATMP
       11401 011410R
C407   11402 016001X       FMP HALF
       11403 000261R
C408   11404 037331R       ISZ .CHEB
```

```
C409  11405 127331R       JMP .CHEB,I   ANS=(D-A)/2
C410*
C411  11406 000000  X2TMP BSS 2
C412  11410 000000  ATMP  BSS 2
C413  11412 000000  BTMP  BSS 2
C414  11414 000000  CTMP  BSS 2
C415  11416 000000  DTMP  BSS 2




C417*           ********************************************
C418*           SUBROUTINE TO COMPUTE THE ENTIER OF A NUMBER
C419*           WHOSE EXPONENT IS LESS THAN 15
C420*           ********************************************
C421*
C422*           CALLING SEQUENCE:
C423*
C424*           LDA X          (FLOATING)
C425*           LDA X+1
C426*           JSB .IENT      (RESULT INTERGER)
C427*           JSB ERROR      (EXIT IF EXPO(X)>14)
C428*
C429*
C430  11420 000000  .IENT NOP
C431  11421 073406R       STA X2TMP     STORE HIGH PART
C432  11422 060001       LDA 1         MOVE LOW PART TO A
C433  11423 010074       AND MSK0      ISOLATE EXPONENT
C434  11424 000033       SLA,RAR
C435  11425 027431R       JMP *+4       IF NEGATIVE OK
C436  11426 042235R       ADA M15
C437  11427 002021       SSA,RSS       EXPO(X) > 14
C438  11430 127420R       JMP .IENT,1   YES, ERROR RETURN
C439  11431 037420R       ISZ .IENT     NO BUMP RETURN POINT
C440  11432 063406R       LDA X2TMP     RESTORE HIGH PART
C441  11433 017160R       JSB IFIX      CALL ENTIER
C442  11434 000000       NOP
C443  11435 060001       LDA 1         PUT RESULT INTO (A)
C444  11436 127420R       JMP .IENT,I
```

```
C446*                ********************************
C447*                SUBROUTINE TO FLOAT AN INTEGER
C448*                ********************************
C449*
C450*                CALLED BY JSB FLOAT WITH INTEGER IN A
C451*                THE FLOATING POINT EQUIVALENT IS RETURNED
C452*                IN A & B
C453*
C454  11437 000000  FLOAT NOP
C455  11440 064070        LDB .15
C456  11441 076042R       STB EXP
C457  11442 005400        CLB
C458  11443 015647R       JSB .PACK
C459  11444 127437R       JMP FLOAT,I




C461*                ****************************************
C462*                SUBROUTINE TO MULTIPLY BY A POWER OF TWO
C463*                ****************************************
C464*
C465*                CALLING SEQUENCE
C466*
C467*                LDA X           (FLOATING)
C468*                LDB X+1
C469*                JSB .PWR2       (RESULT FLOATING)
C470*                DEC N           (INTEGER POWER)
C471*
C472*                RETURNS WITH X*2^N IN A&B
C473*                NO CHECK IS MADE FOR EXPONENT
C474*                OVERFLOW OR UNDERFLOW
C475*
C476  11445 000000  .PWR2 NOP
C477  11446 002003        SZA,RSS         X=0 ?
C478  11447 027461R       JMP .RET        YES, ANS=0
C479  11450 073406R       STA X2TMP
C480  11451 017223R       JSB .FLUN
C481  11452 077407R       STB X2TMP+1
C482  11453 143445R       ADA .PWR2,I
C483  11454 001200        RAL
C484  11455 010074        AND MSK0        NEW EXPO = (OLD EXPO) +N
C485  11456 070001        STA 1
C486  11457 047407R       ADB X2TMP+1     KEEP OLD MANTISSA
C487  11460 063406R       LDA X2TMP
C488  11461 037445R .RET  ISZ .PWR2
C489  11462 127445R       JMP .PWR2,I


C491  11463 000000  TT1   BSS 1
C492  11464 000000  TT2   BSS 1
C493  00051         TT3   EQU TEMPS+4
C494  00052         TT4   EQU TEMPS+5
C495  11057         FFLAG EQU SBUX
```

```
C497*                ******************************
C498*                MATRIX STMT EXECUTION CONTROL
C499*                ******************************
C500   11465 017062R EMAT   JSB LDTMP
C501   11466 036045R        ISZ TEMPS      MAT READ
C502   11467 012173R        AND MSK1A :::
C503   11470 002002         SZA                 MAT PRINT?
C504   11471 027620R        JMP EMAT7      NO
C505   11472 017062R        JSB LDTMP     YES
C506   11473 012220R        AND OPMSK      SAVE
C507   11474 072057R        STA MLBX1        TYPE
C508   11475 052213R        CPA RDOP       PRINT?
C509   11476 002001         RSS            NO
C510   11477 016671R        JSB PRNIN        YES
C511*
C512   11500 017062R EMAT1  JSB LDTMP    LOAD
C513   11501 012176R        AND MSK1         OPERAND
C514   11502 002003         SZA,RSS        NULL? (END OF MAT PRINT)
C515   11503 026235R        JMP XEC4           YES
C516   11504 017613R        JSB SSYMT        NO, SEARCH THE SYMBOL TABLE
C517   11505 006007         INB,SZB,RSS    FOUND?
C518   11506 027302R        JMP E8-1         NO
C519   11507 036045R        ISZ TEMPS      YES
C520   11510 160001         LDA 1,I        SAVE ARRAY
C521   11511 012222R        AND INF
C522   11512 072061R        STA B1           BASE ADDRESS
C523   11513 062057R        LDA MLBX1
C524   11514 052213R        CPA RDOP       READ?
C525   11515 027571R        JMP EMAT5      YES
C526   11516 044055         ADB .2         NO
C527   11517 160001         LDA 1,I        SAVE
C528   11520 072062R        STA B1+1         DIMENSIONS
C529   11521 010074         AND MSK0       SET
C530   11522 003004         CMA,INA          COLUMN
C531   11523 072063R        STA B2             COUNTERS
C532   11524 072064R        STA B2+1
C533   11525 160001         LDA 1,I        SET
C534   11526 001727         ALF,ALF
C535   11527 010074         AND MSK0         ROW
C536   11530 003004         CMA,INA
C537   11531 072065R        STA B3             COUNTER
C538   11532 016076R        JSB LCHK2        SET DELIMITER
C539   11533 002400         CLA            SET DELIMITER
C540   11534 072001R        STA MCKS         AS COMMA
C541   11535 062045R        LDA TEMPS      MORE
C542   11536 052031R        CPA PRADD        STATEMENT?
C543   11537 027550R        JMP EMAT3      NO
C544   11540 017062R        JSB LDTMP    YES
C545   11541 012220R        AND OPMSK      EXTRACT DELIMITER
C546   11542 052200R        CPA B1400    ::
C547   11543 036001R        ISZ MCKS       YES
C548   11544 027550R        JMP EMAT3
C549   11545 006400  EMAT2  CLB            COMMA
C550   11546 056001R        CPB MCKS         DELIMETER?
C551   11547 017440R        JSB EDELM      YES
```

```
C552  11550 017322R EMAT3 JSB ODLD      LOAD
C553  11551 000061R       DEF B1          NEXT
C554  11552 036061R       ISZ B1             ELEMENT
C555  11553 036061R       ISZ B1
C556  11554 017426R       JSB ENOUT     OUTPUT IT
C557  11555 036063R       ISZ B2        ROW COMPLETE?
C558  11556 027545R       JMP EMAT2     NO
C559  11557 017461R       JSB OUTLN     YES, DO
C560  11560 017461R       JSB OUTLN       SPACING
C561  11561 062064R       LDA B2+1      RESET
C562  11562 072063R       STA B2          COLUMN COUNTER
C563  11563 036065R       ISZ B3        ARRAY EXHAUSTED?
C564  11564 027550R       JMP EMAT3     NO
C565  11565 066045R EMAT4 LDB TEMPS     YES, MORE
C566  11566 056031R       CPB PRADD       STATEMENT?
C567  11567 026233R       JMP XEC4        NO
C568  11570 027500R       JMP EMAT1     YES
C569*
C570  11571 076063R EMAT5 STB B2        SAVE SYMBOL TABLE POINTER
C571  11572 017062R       JSB LDTMP    EXTRACT
C572  11573 012220R       AND OPMSK       NEXT OPERATOR
C573  11574 066045R       LDB TEMPS     STATEMENT
C574  11575 056031R       CPB PRADD       EXHAUSTED?
C575  11576 002400        CLA           YES
C576  11577 052206R       CPA B1100   ::
C577  11600 017744R       JSB REDIM     YES, REDIMENSION ARRAY
C578  11601 062063R       LDA B2        LOAD
C579  11602 040055        ADA .2          ARRAY
C580  11603 160000        LDA 0,I           DIMENSIONS
C581  11604 017130R       JSB MDIM      SET
C582  11605 001100        ARS             ARRAY
C583  11606 003004        CMA,INA           ELEMENT
C584  11607 072065R       STA B3              COUNTER
C585  11610 016524R EMAT6 JSB FDATA       FETCH VALUE
C586  11611 017342R       JSB ODST      STORE
C587  11612 000061R       DEF B1
C588  11613 036061R       ISZ B1        IT
C589  11614 036061R       ISZ B1
C590  11615 036065R       ISZ B3        ARRAY EXHAUSTED?
C591  11616 027610R       JMP EMAT6     NO
C592  11617 027565R       JMP EMAT4     YES
C593*
C594  11620 017613R EMAT7 JSB SSYMT       SAVE
C595  11621 006004        INB             BASE ADDRESS
C596  11622 160001        LDA 1,I           OF .DESTINATION
C597  11623 012222R       AND INF
C598  11624 072065R       STA B3              ARRAY
C599  11625 076063R       STB B2        SAVE SYMBOL TABLE ADDRESS
C600  11626 044055        ADB .2        SAVE
C601  11627 160001        LDA 1,I         ITS
C602  11630 072066R       STA B3+1          DIMENSIONS
C603  11631 002404        CLA,INA       ASSUME MAT
C604  11632 073440R       STA EDELM       REPLACEMENT
C605  11633 017062R EMAT0 JSB LDTMP  LOAD NEXT
C606  11634 036045R       ISZ TEMPS       OPERAND
```

```
C607  11635 002020          SSA             ARRAY FUNCTION?
C608  11636 027723R         JMP EMA11       YES
C609  11637 012176R EMAT8   AND MSK1        NO
C610  11640 002003          SZA,RSS         SCALAR MULTIPLICATION?
C611  11641 027713R         JMP EMA10       YES
C612  11642 017613R         JSB SSYMT         NO
C613  11643 006004          INB             SAVE
C614  11644 160001          LDA 1,I           BASE
C615  11645 012222R         AND INF
C616  11646 072061R         STA B1              ADDRESS AND
C617  11647 044055          ADB .2                DIMENSIONS
C618  11650 160001          LDA 1,I                 OF FIRST
C619  11651 072062R         STA B1+1                  SOURCE ARRAY
C620  11652 066045R         LDB TEMPS       STATEMENT
C621  11653 056031R         CPB PRADD         EXHAUSTED?
C622  11654 027674R         JMP EMAT9       YES
C623  11655 017062R         JSB LDTMP  NO EXTRACT
C624  11656 001727          ALF,ALF           AND
C625  11657 010072          AND .63             RECORD
C626  11660 040045          ADA M6                EMAT
C627  11661 073440R         STA EDELM               OPERATOR
C628  11662 017062R         JSB LDTMP    SAVE
C629  11663 012176R         AND MSK1
C630  11664 017613R         JSB SSYMT         BASE ADDRESS
C631  11665 006004          INB
C632  11666 160001          LDA 1,I             AND DIMENSIONS
C633  11667 012222R         AND INF
C634  11670 072063R         STA B2
C635  11671 044055          ADB .2                OF SECOND
C636  11672 160001          LDA 1,I
C637  11673 072064R         STA B2+1                SOURCE ARRAY
C638*
C639  11674 073440R EMAT9   LDA EDELM      TRANSFER TO
C640  11675 043700R         ADA LMAP          APPROPRIATE
C641  11676 114000          JSB 0,I             ROUTINE
C642  11677 026233R         JMP XEC4
C643*
C644  11700 111700R LMAP    DEF LBASE-1,I
C645  11701 012151R LBASE   DEF REPLC
C646  11702 012134R         DEF ADD
C647  11703 012144R         DEF SUB
C648  11704 012514R         DEF MULT
C649  11705 012212R         DEF SZER
C650  11706 012167R         DEF LCON
C651  11707 012220R         DEF LIDN
C652  11710 012424R         DEF LINV
C653  11711 012252R         DEF TRAN
C654  11712 012162R         DEF SMULT
C655*
C656  11713 000065  EMA10   LDA .10        SET SMULT
C657  11714 073440R         STA EDELM       OPERATOR
C658  11715 015255R         JSB FETCH       EVALUATE
C659  11716 072057R         STA MLBX1       AND SAVE
C660  11717 075060R         STB MLBX1+1       SCALAR
C661  11720 035045R         ISZ TEMPS       GO TO
```

```
C662  11721 036045R        ISZ TEMPS        PROCESS
C663  11722 027633R        JMP EMAT0          SOURCE ARRAY
C664*
C665  11723 001727  EMA11  ALF,ALF         EXTRACT
C666  11724 001722         ALF,RAL  :::
C667  11725 010071         AND .31          TYPE
C668  11726 040043         ADA M8         RECORD EMAT
C669  11727 073440R        STA EDELM        OPERATOR TYPE
C670  11730 040043         ADA M8         INV OR
C671  11731 002020         SSA              TRN?
C672  11732 027737R        JMP EMA12      NO
C673  11733 017062R        JSB LDTMP    YES, LOAD
C674  11734 036045R        ISZ TEMPS        SOURCE
C675  11735 036045R        ISZ TEMPS          ARRAY
C676  11736 027657R        JMP EMAT8            SYMBOL
C677*
C678  11737 066045R EMA12  LDB TEMPS      REDIMENSIONING
C679  11740 055031R        CPB PRADD        PART?
C680  11741 027674R        JMP EMAT9      NO
C681  11742 017744R        JSB REDIM      YES
C682  11743 027674R        JMP EMAT9
```

```
C684*                ********************************
C685*                SUBROUTINE TO REDIMENSION ARRAY
C686*                ********************************
C687  11744 000000  REDIM NOP
C688  11745 016001R        JSB MCKS       EVALUATE
C689  11746 005727         BLF,BLF          AND SAVE
C690  11747 075066R        STB B3+1           ROW COUNT
C691  11750 006404         CLB,INB        LOAD DEFAULT COLUMN COUNT
C692  11751 036045R        ISZ TEMPS      SINGLE
C693  11752 017062R        JSB LDTMP       DIMENSION
C694  11753 012220R        AND OPMSK          ARRAY?
C695  11754 052203R        CPA B2400   ::
C696  11755 027760R        JMP REDI1      YES
C697  11756 016001R        JSB MCKS       NO, EVALUATE COLUMN COUNT
C698  11757 036045R        ISZ TEMPS      MOVE PAST
C699  11760 036045R REDI1  ISZ TEMPS        RIGHT BRACKET
C700  11761 046066R        ADB B3+1       PACK
C701  11762 076066R        STB B3+1         DIMENSIONS
C702  11763 062063R        LDA B2         STORE IN
C703  11764 040055         ADA .2           SYMBOL
C704  11765 174000         STB 0,I            TABLE
C705  11766 040052         ADA M1         COMPUTE
C706  11767 160000         LDA 0,I          PHYSICAL
C707  11770 017130R        JSB MDIM           ARRAY SPACE
C708  11771 072060R        STA MLBX1+1          SIZE
C709  11772 062066R        LDA B3+1       COMPUTE
C710  11773 017130R        JSB MDIM         NEW SIZE
C711  11774 003004         CMA,INA        NEW
C712  11775 042060R        ADA MLBX1+1      SIZE
```

```
C713  11776  002020          SSA                    ACCEPTABLE?
C714  11777  016274R         JSB ERROR       NO
C715  12000  127744R E7      JMP REDIM,I     YES
```

```
C717*                ******************************************
C718*                SUBROUTINE TO EVALUATE & CHECK A SUBSCRIPT
C719*                ******************************************
C720   12001 000000  MCKS  NOP
C721   12002 016255R       JSB FETCH        CALL FOR EVALUATION
C722   12003 017147R       JSB SBFIX      CONVERT TO INTEGER (ROUNDED)
C723   12004 005004        INB            UNBIAS SUBSCRIPT
C724   12005 060001        LDA 1          PUT INTO (A)
C725   12006 040075        ADA M256       LESS THAN
C726   12007 002021        SSA,RSS          256?
C727   12010 027221R       JMP E6-1          NO
C728   12011 125001R       JMP MCKS,I     YES, RETURN SUBSCRIPT IN (B)
```

```
C730*
C731*************************************************
C732******      MATRIX   ROUTINES          ********
C733*************************************************
C734*CALL FOR MATRIX OPERATION IS MADE WITH FOUR*
C735*PARAMETERS,ROUTINE NUMBER AND ADDRESS OF    *
C736*SYMBOL TABLE OF THREE MATRICES. FOR SCALAR *
C737*MULT,LAST IS ADDRESS OF SCALAR VALUE        *
C738*OPERATION IS OF FORM  B3=B1 OP B2           *
C739*THE ADDRESS OF THE BASE ADDRESS OF MATRICES*
C740*IS GIVEN IN B1,B2,B3. THE DIMENSIONS OF A  *
C741*MATRIX IS GIVEN IN B(I)+1, ROWS IN MOST SIG*
C742*PART(MSP) AND COLUMN IN LEAST SIG PART(LSP)*
C743*************************************************
C744*
C745*
C746*
C747*************************************************
C748***       SUBROUTINE  GENERAL                 ***
C749*************************************************
C750*B3=B1 OP B2  SUBROUTINE COMPUTES AN ELEMENT*
C751*OF B3 AND INCREMENTS TO NEXT ELEMENT. THE   *
C752*OPERATION THAT IS PERFORMED AND              *
C753* THE MATRICES INCREMENTED ARE                *
C754*  MODIFIED BY ROUTINES ADD, SUB, REPL        *
C755*SCALAR MULT, CON,ZERO,IDN. ROUTINE CHECKS   *
C756*COMPATIBILITY OF THREE MATRICES USING SUB   *
C757*COMPARE (PARAMETERS SUPPLIED IN REG A,B)    *
C758*************************************************
C759*
C760  12012 000000  GENER NOP             SUBROUTINE GENERAL
C761  12013 062064R       LDA B2+1        LOAD DIM FOR MATRIX 2
C762  12014 066062R       LDB B1+1        LOAD DIM FOR MATRIX 1
C763  12015 015063R       JSB COMPR       CHECKS ROW AND COL DIM
C764*                                     ARE COMPATIBLE
C001  12016 062062R GEN2  LDA B1+1        LOAD DIM FOR MATRIX 1
C002  12017 066066R       LDB B3+1        LOAD DIM FOR MATRIX 3
C003  12020 015063R       JSB COMPR       CHECK ROW AND COL DIM
C004  12021 100200        MPY T3      COLUMNS IN (A), ROWS IN T3
      12022 013273R
C005  12023 003004        CMA,INA
C006  12024 073512R       STA LPIV        -ROWS*COLUMNS
C007*                                     COMPUTES B3=B1 OP B2
C008  12025 017322R LOOP  JSB DDLD        LOAD
C009  12026 000061R       DEF B1              NEXT
C010  12027 056061R       ISZ B1                SOURCE
C011  12030 056061R       ISZ B1                     ELEMENT
C012  12031 126032R       JMP *+1,I
C013  12032 000000  MOD   BSS 1
C014  12033 017342R       JSB DDST        STORE
C015  12034 000065R       DEF B3              NEXT
C016  12035 056065R       ISZ B3                DESTINATION
C017  12036 056065R       ISZ B3                           ELEMENT
C018  12037 060046        LDA B5
C019  12040 042062R       ADA MOD2
```

```
C020  12041 002021        SSA,RSS
C021  12042 026045R       JMP *+3
C022  12043 035063R       ISZ B2
C023  12044 035063R       ISZ B2
C024  12045 037312R       ISZ LP1V
C025  12046 026025R       JMP LOOP       COMPUTE NEXT ELEMENT
C026  12047 126012R       JMP GENER,I
C027  12050 012050R MOD1  DEF *
C028  12051 017403R       JSB .FAD.
C029  12052 100063R       DEF B2,I
C030  12053 026033R       JMP MOD+1
C031  12054 017412R       JSB .FSB.
C032  12055 100063R       DEF B2,I
C033  12056 026033R       JMP MOD+1
C034  12057 017421R       JSB .FMP.
C035  12060 000123R       DEF MBXL
C036  12061 026033R       JMP MOD+1
C037  12062 000000  MOD2  BSS 1
C038*
C039*
C040*********************************************
C041****     SUBROUTINE COMPARE                ****
C042*********************************************
C043*ROUTINE COMPARES DIM OF TWO MATRICES GIVEN *
C044*THEIR DIM IN REGISTERS A,B                 *
C045*DIMENSIONS ARE GIVEN IN B(I)+2             *
C046*********************************************
C047*
C048  12063 000000  COMPR NOP
C049  12064 050001        CPA 1          EQUAL?
C050  12065 002001        RSS            YES
C051  12066 016274R LERR  JSB ERROR      NO
C052  12067 001727        ALF,ALF        SAVE
C053  12070 010074        AND MSK0         # OF
C054  12071 073273R       STA T3             ROWS
C055  12072 060001        LDA 1
C056  12073 010074        AND MSK0       SAVE #
C057  12074 073274R       STA T4           OF COLUMNS
C058  12075 126063R       JMP COMPR,I
C059*
C060*
C061*********************************************
C062******     SUBROUTINE   LCHK             ******
C063*********************************************
C064*TESTS THAT NO ELEMENT IN A MATRIX IS       *
C065*UNASSIGNED. ENTRY1 CHECKS MATRICES GIVEN BY*
C066*B1 AND B2 AND ENTRY 2 CHECKS ONLY B1       *
C067*********************************************
C068*
C069  12076 000000  LCHK2 NOP
C070  12077 062076R       LDA LCHK2
C071  12100 072102R       STA LCHK1
C072  12101 026106R       JMP *+5
C073  12102 000000  LCHK1 NOP
C074  12103 066063R       LDB B2         BASE ADDR
```

```
C075  12104 062064R        LDA B2+1       ROW AND COL DIM.
C076  12105 015112R        JSB LCHK4      TEST EACH TERM OF B2
C077  12106 065061R        LDB B1         BASE ADDR
C078  12107 062062R        LDA B1+1       ROW AND COL DIM.
C079  12110 015112R        JSB LCHK4      TEST EACH TERM OF B1
C080  12111 126102R        JMP LCHK1,I
C081*
C082  12112 000000  LCHK4  NOP            SUBROUTINE TO TEST TERMS
C083  12113 077276R        STB I6         SAVE
C084  12114 017150R        JSB MDIM       COMPUTE SIZE OF MATRIX
C085  12115 001100          ARS            SET NEGATIVE
C086  12116 003004          CMA,INA
C087  12117 073277R        STA I7         COUNTER FOR ELEMENTS
C088  12120 017522R LCHK6  JSB DDLD         LOAD
C089  12121 013276R        DEF I6
C090  12122 037276R        ISZ I6              ELEMENT
C091  12123 037276R        ISZ I6
C092  12124 052263R        CPA MNEG       COMPARE WITH PRESET QTY.
C093  12125 025127R        JMP *+2
C094  12126 025131R        JMP LCHK5
C095  12127 052264R        CPB MNEG+1
C096  12130 016274R        JSB ERROR      ERROR 'MAT UNASSIGNED'
C097  12131 037277R LCHK5  ISZ I7         DONE?
C098  12132 025120R        JMP LCHK6      NO
C099  12133 126112R        JMP LCHK4,I    YES
C100*
C101*
C102**********************************************
C103****  SUBROUTINE  MATRIX  ADD              ****
C104**********************************************
C105*B1,B2,B3 CONTAIN ADDRESS OF BASE ADDRESS OF*
C106*THREE MATRICES. ROUTINE EXECUTES B3=B1+B2  *
C107*BY MODIFYING INSTR IN ROUTINE GENERAL       *
C108**********************************************
C109*
C110  12134 000000  ADD    NOP
C111  12135 060054          LDA .1
C112  12136 072062R ADD1   STA MOD2
C113  12137 042050R        ADA MOD1
C114  12140 072032R        STA MOD
C115  12141 016102R        JSB LCHK1      TEST B1,B2 FOR UNASSIGNED TERMS
C116  12142 016012R        JSB GENER      ROUTINE GENERAL
C117  12143 126134R        JMP ADD,I      EXIT TO MAIN PROGRAM
C118*
C119*
C120**********************************************
C121****  SUBROUTINE  MATRIX  SUBTRACT         ****
C122**********************************************
C123*B1,B2,B3 CONTAIN ADDRESS OF BASE ADDRESS OF*
C124*THREE MATRICES. ROUTINE EXECUTES B3=B1-B2  *
C125*BY MODIFYING INSTR IN ROUTINE GENERAL       *
C126**********************************************
C127*
C128  12144 000000  SUB    NOP            LET
C129  12145 062144R        LDA SUB          ADD DO
```

```
C130   12146 072134R        STA ADD          RETURN
C131   12147 060057         LDA .4
C132   12150 025136R        JMP ADD1
```

```
C134*
C135*********************************************
C136****   SUBROUTINE  MATRIX  REPLACE        ****
C137*********************************************
C138*B1,B3 GIVE ADDRESS OF BASE ADDRESS OF GIVEN*
C139*MATRIX AND RECEIVING MATRIX RESPECTIVELY   *
C140*********************************************
C141   12151 000000  REPLC NOP             LET
C142   12152 062151R       LDA REPLC         GENER DO
C143   12153 072012R       STA GENER           RETURN
C144   12154 060064        LDA .9
C145   12155 072062R REPL1 STA MOD2
C146   12156 042050R       ADA MOD1
C147   12157 072032R       STA MOD
C148   12160 016076R       JSB LCHK2       TEST B1 FOR UNASSIGNED ELEMENTS
C149   12161 026016R       JMP GEN2
C150*
C151*
C152*********************************************
C153****  SUBROUTINE MATRIX SCALAR MULT       ****
C154*********************************************
C155*B1,B3 GIVE ADDRESS OF BASE ADDRESS OF GIVEN*
C156*MATRIX AND RECEIVING MATRIX RESPECTIVELY   *
C157*MBXL HOLDS ADDRESS OF SCALAR VALUE         *
C158*********************************************
C159*
C160   12162 000000  SMULT NOP             LET
C161   12163 062162R       LDA SMULT         GENER DO
C162   12164 072012R       STA GENER           RETURN
C163   12165 060062        LDA .7
C164   12166 026155R       JMP REPL1
C165*
C166*
C167*********************************************
C168****    SUBROUTINE  MATRIX  CON           ****
C169*********************************************
C170*SETS MATRIX TO ALL ONES.  B3 IS ADDRESS OF *
C171*BASE ADDRESS OF MATRIX.                    *
C172*********************************************
C173*
C174   12167 000000  LCON  NOP
C175   12170 062261R       LDA HONE
C176   12171 064055        LDB .2
C177   12172 072057R LCON1 STA MLBX1       SET
C178   12173 076060R       STB MLBX1+1       CONSTANT
C179   12174 062066R       LDA B3+1
C180   12175 017130R       JSB MDIM        SET
C181   12176 001100        ARS               ELEMENT
C182   12177 003004        CMA,INA             COUNTER
C183   12200 073512R       STA LPIV
C184   12201 062057R       LDA MLBX1       LOAD
C185   12202 066060R       LDB MLBX1+1       CONSTANT
C186   12203 017342R LCON2 JSB DDST          STORE
C187   12204 000065R       DEF B3                IN
C188   12205 036065R       ISZ B3                 NEXT
```

```
C189  12206 035065R         ISZ B5                    ELEMENT
C190  12207 037512R         ISZ LPIV        DONE?
C191  12210 026203R         JMP LCON2       NO
C192  12211 125167R         JMP LCON,I      YES
C193*
C194*
C195*********************************************
C196****    SUBROUTINE  MATRIX  ZERO           ****
C197*********************************************
C198*SETS MATRIX TO ZERO. B3 IS ADDRESS OF BASE *
C199*ADDRESS OF MATRIX.  B1,B2 ARE REDUNDANT    *
C200*SET B1=0 AND USE SUBROUTINE CON,ENTRY2      *
C201*********************************************
C202*
C203  12212 000000  SZER  NOP
C204  12213 062212R         LDA SZER        CONVERT
C205  12214 072167R         STA LCON
C206  12215 002400         CLA               LCON
C207  12216 005400         CLB
C208  12217 026172R         JMP LCON1           TO SZER
C209*
C210*
C211*********************************************
C212****    SUBROUTINE  MATRIX  IDN            ****
C213*********************************************
C214*ROUTINE SETS UP IDENTITY MATRIX             *
C215*B3 IS ADDRESS OF BASE ADDRESS OF MATRIX     *
C216*USE SZER TO SET MATRIX TO ALL ZEROS.  ON    *
C217*RETURN CHECK FOR SQUARE MATRIX.             *
C218*********************************************
C219*
C220  12220 000000  LIDN  NOP
C221  12221 062065R         LDA B3
C222  12222 073501R         STA T9          SAVE BASE ADDRESS
C223  12223 016212R         JSB SZER        SET ALL MATRIX TO ZERO
C224  12224 062066R         LDA B3+1        IS
C225  12225 001727         ALF,ALF           ARRAY
C226  12226 052066R         CPA B3+1            SQUARE?
C227  12227 001010         ALS,SLA         YES
C228  12230 026066R         JMP LERR        NO
C229  12231 012176R         AND MSK1        SAVE ROW
C230  12232 072057R         STA MLBX1         LENGTH
C231  12233 001100         ARS             SAVE
C232  12234 003004         CMA,INA           ROW
C233  12235 072060R         STA MLBX1+1         COUNTER
C234  12236 057501R         LDB T9          RESTORE
C235  12237 075065R         STB B3            B3
C236  12240 062261R LIDN1  LDA HONE        STORE
C237  12241 017055R         JSB STBI
C238  12242 006004         INB               1.0 ON
C239  12243 060055         LDA .2
C240  12244 017055R         JSB STBI              DIAGONAL
C241  12245 006004         INB
C242  12246 046057R         ADB MLBX1       MOVE TO NEXT DIAGONAL ELEMENT
C243  12247 035060R         ISZ MLBX1+1     DONE?
```

```
0244   12250  025240R        JMP LIDN1      NO
0245   12251  126220R        JMP LIDN,I     YES
```

```
C247*
C248*************************************************
C249****      SUBROUTINE  TRANSPOSE          *****
C250*************************************************
C251*TRANSPOSE OF FORM B3(M,N)=T(B1(N,M))        *
C252*B1,B3 GIVE ADDRESS OF BASE ADDRESS OF GIVEN*
C253*AND RECEIVING MATRICES RESPECTIVELY.       *
C254*************************************************
C255*
C256  12252 000000  TRAN  NOP
C257  12253 016076R        JSB LCHK2      TEST B1 FOR UNASSIGNED TERMS
C258*                                     CHECK DIMENSIONS
C259  12254 062066R        LDA B3+1       PARAMETERS OF B3
C260  12255 001727        ALF,ALF        INTERCHANGE ROW AND COLUMN
C261  12256 066062R        LDB B1+1       PARAMETERS OF B1
C262  12257 016063R        JSB COMPR      SUBROUTINE COMPARE
C263*                                     DIMENSIONS COMPATIBLE
C264  12260 100200        MPY T3      #OF COLUMNS IN (A), # OF ROWS IN T3
      12261 013273R
C265  12262 073312R        STA LP1V       PRODUCT OF ROW*COL
C266  12263 063274R        LDA T4         SET
C267  12264 003004        CMA,INA          COLUMN
C268  12265 073275R        STA T5             COUNTER
C269*                                     T6 IS INDICATOR TO SELECT
C270*                                     WHICH ELEMENT IN A COL OF
C271*                                     B1 IS TO BE TRANSPOSED
C272  12266 002400  TRAN1 CLA
C273  12267 073276R        STA T6         SET T6=0
C274  12270 067276R LNEXT LDB T6         LOAD
C275  12271 005000        BLS
C276  12272 046061R        ADB B1           NEXT ELEMENT
C277  12273 060001        LDA 1
C278  12274 017505R        JSB FTHOP            OF COLUMN
C279  12275 017542R        JSB DDST          STORE
C280  12276 000065R        DEF B3                IN
C281  12277 056065R        ISZ B3                  ROW
C282  12300 056065R        ISZ B3
C283  12301 063276R        LDA T6         SET T6=T6+T4
C284  12302 043274R        ADA T4         T6 POINTS TO NEXT TERM IN
C285  12303 073276R        STA T6         A COLUMN TO BE TRANSPOSED
C286  12304 053312R        CPA LP1V       TEST FOR LAST IN COL
C287  12305 026307R        JMP *+2
C288  12306 026270R        JMP LNEXT
C289*                                     SET BASE ADDRESS TO FIRST
C290*                                     TERM IN NEXT COLUMN
C291  12307 056061R        ISZ B1
C292  12310 056061R        ISZ B1
C293  12311 057275R        ISZ T5
C294  12312 026266R        JMP TRAN1      TRANSPOSE NEXT COL
C295  12313 126252R        JMP TRAN,I     EXIT TO MAIN PROGRAM
```

```
C297*
C298*********************************************
C299****    SUBROUTINE  MATRIX  MULT          ****
C300*********************************************
C301*ROUTINE IS OF FORM B3(M,P)=B1(M,N)*B2(N,P) *
C302*B1,B2,B3 ARE ADDRESSES OF BASE ADDRESSES OF*
C303*THREE MATRICES                             *
C304*********************************************
C305*
C306  12314 000000  MULT  NOP
C307  12315 015102R       JSB LCHK1      TEST B1,B2 FOR UNASSIGNED TERMS
C308*                                    CHECK DIMENSIONS
C309  12316 062066R       LDA B3+1       PARAMETERS OF B3
C310  12317 010074       AND MSK0       SAVE COLUMN COUNT
C311  12320 073276R       STA T6
C312  12321 062064R       LDA B2+1       PARAMETERS OF B2
C313  12322 010074       AND MSK0
C314  12323 053276R       CPA T6         COLUMNS EQUAL
C315  12324 002001       RSS              IN NUMBER?
C316  12325 026066R       JMP LERR       NO
C317*                                    COMBINE B3,B2 PARAMETERS
C318*                                    INTO (M,N) AND COMPARE
C319*                                    WITH THOSE OF B1
C320  12326 062066R       LDA B3+1       PARAMETERS OF B3
C321  12327 010075       AND M256
C322  12330 070001       STA 1          STORE ROW IN MSP OF B
C323  12331 062064R       LDA B2+1       PARAMETERS OF B2
C324  12332 001727       ALF,ALF        GET ROW COUNT
C325  12333 010074       AND MSK0         IN (A)
C326  12334 040001       ADA 1          COMBINE A AND B
C327  12335 066062R       LDB B1+1       PARAMETERS OF B1
C328  12336 015063R       JSB COMPR      COMPARE ROW AND COL
C329*                                    DIMENSIONS ARE COMPATIBLE
C330*                                    M,N ARE STORED IN T3,T4
C331*                                    SAVE B2 AS .DESTROYED IN
C332  12337 062063R       LDA B2         MULT
C333  12340 073275R       STA T5
C334  12341 063273R       LDA T3         SET
C335  12342 003004       CMA,INA          ROW
C336  12343 073301R       STA I9             COUNTER
C337  12344 063276R MULT4 LDA T6
C338  12345 003004       CMA,INA
C339  12346 073302R       STA I10        SET COUNTER
C340  12347 063275R       LDA T5
C341  12350 072063R       STA B2         RESTORE BASE ADDRESS B2
C342  12351 002400  MULT3 CLA
C343  12352 073303R       STA I11        COUNTER FOR B2. INCR BY
C344*                                    2*P AND POINTS TO NEXT TERM
C345*                                    IN COL TO BE MULTIPLIED
C346  12353 073304R       STA I12        COUNTER FOR B1. INCR BY 2
C347*                                    AND POINTS TO NEXT TERM
C348*                                    IN ROW TO BE MULTIPLIED
C349  12354 005400       CLB
C350  12355 017342R       JSB DDST        CLEAR TO ZERO
C351  12356 000065R       DEF B3
```

```
C352  12357 065061R MULT2 LDB B1        COMPUTE PROD OF ONE TERM
C353  12360 047304R       ADB T12       IN ROW BY ONE TERM IN COL
C354  12361 077310R       STB T18
C355  12362 065063R       LDB B2
C356  12363 047303R       ADB T11
C357  12364 060001        LDA B
C358  12365 017505R       JSB FTHOP
C359  12366 017421R       JSB .FMP.
C360  12367 113510R       DEF T18,I
C361  12370 017403R       JSB .FAD.     COMPUTES RUNNING SUM
C362  12371 100065R       DEF B3,I
C363  12372 017542R       JSB DDST
C364  12373 000065R       DEF B3
C365  12374 037304R       ISZ T12       SELECT NEXT TERM IN ROW
C366  12375 037304R       ISZ T12
C367  12376 063276R       LDA T6        SELECT NEXT TERM IN COL
C368  12377 001000        ALS
C369  12400 043303R       ADA T11
C370  12401 073303R       STA T11
C371*                                   TEST IF HAVE MULT ONE ROW
C372*                                   BY ONE COLUMN
C373  12402 063274R       LDA T4
C374  12403 001000        ALS
C375  12404 053304R       CPA T12
C376  12405 026407R       JMP *+2
C377  12406 026357R       JMP MULT2     MULT AND ADD IN NEXT TERM
C378*                                   SUMMATION OF PRODUCTS FOR
C379*                                   ONE TERM OF B3 IS DONE
C380*                                   MULT SAME ROW BY NEXT COL
C381  12407 036065R       ISZ B3        INCR RECEIVING MAT
C382  12410 036065R       ISZ B3
C383  12411 036063R       ISZ B2        BASE ADDRESS OF NEXT COL
C384  12412 036063R       ISZ B2
C385*                                   TEST IF HAVE MULT ONE ROW
C386*                                   BY ALL COLUMNS
C387  12413 037302R       ISZ T10       SKIP IF INNERPRODUCT DONE
C388  12414 026351R       JMP MULT3     COMPUTE SAME ROW*NEXT COL
C389*                                   SELECT NEXT ROW
C390  12415 063274R       LDA T4
C391  12416 001000        ALS
C392  12417 042061R       ADA B1
C393  12420 072061R       STA B1        ADDRESS OF NEXT ROW
C394  12421 037301R       ISZ T9
C395  12422 026344R       JMP MULT4     MULT ROW BY ALL COLUMNS
C396  12423 126314R       JMP MULT,I    EXIT TO MAIN PROGRAM
```

```
C398*
C399*************************************************
C400****    SUBROUTINE  MATRIX  INVERT          ****
C401*************************************************
C402*OPERATION OF FORM  MAT B3 = INV B1             *
C403*B1,B3 ARE ADDRESSES OF BASE ADDRESS OF        *
C404*MATRIX TO BE INVERTED AND RECEIVING MATRIX    *
C405*RESPECTIVELY. B2 IS REDUNDANT. METHOD USED    *
C406*IS GAUSSIAN ELIMINATION WITH COLUMN           *
C407*PIVOTING                                      *
C408*************************************************
C409*
C410  12424 000000  LINV  NOP             SUBROUTINE MATRIX INVERT
C411  12425 015076R       JSB LCHK2       TEST B1 FOR UNASSIGNED TERMS
C412  12426 062062R       LDA B1+1        DIMENSIONS OF MATRIX B1
C413  12427 065066R       LDB B3+1        DIMENSIONS OF MATRIX B3
C414  12430 015063R       JSB COMPR       CHECK DIMENSIONS
C415*                                     ROW AND COL VALUES T3,T4
C416*                                     MAKE COPY OF MATRIX B1
C417*                                        IN FREE CORE
C418  12431 062065R       LDA B3          SAVE
C419  12432 073305R       STA T13           B3
C420  12433 062062R       LDA B1+1        COMPUTE SIZE
C421  12434 017130R       JSB MDIM          OF MATRIX
C422  12435 003004       CMA,INA
C423  12436 001100       ARS             SAVE
C424  12437 073272R       STA T2            ELEMENT
C425  12440 001000       ALS                 COUNTER
C426  12441 065007R       LDB FDISK       SAVE
C427  12442 006004       INB               ADDRESS OF
C428  12443 075063R       STB B2              FREE CORE
C429  12444 075065R       STB B3
C430  12445 007004       CMB,INB         COMPUTE SIZE OF
C431  12446 045003R       ADB LWAM          FREE CORE AREA
C432  12447 040001       ADA 1           ENOUGH
C433  12450 002020       SSA               CORE LEFT?
C434  12451 027240R       JMP E1          NO
C435  12452 016151R       JSB REPLC       YES, COPY SOURCE MATRIX
C436  12453 063305R       LDA T13         RESTORE
C437  12454 072065R       STA B3            B3
C438  12455 016220R       JSB LIDN        SET .DESTINATION TO IDENTITY
C439  12456 063305R       LDA T13         RESTORE ITS
C440  12457 072065R       STA B3            BASE ADDRESS
C441  12460 002400       CLA             INITIALIZE
C442  12461 073304R       STA T12           MAXIMUM
C443  12462 073305R       STA T13             ELEMENT
C444  12463 062063R       LDA B2          COPY B2 INTO B1 AS
C445  12464 072061R       STA B1          B2 NEEDED LATER
C446  12465 017322R LIN11 JSB DDLD         LOAD
C447  12466 000061R       DEF B1              NEXT
C448  12467 036061R       ISZ B1                 ELEMENT
C449  12470 036061R       ISZ B1
C450  12471 036061R       ISZ B1
C451  12472 002020       SSA               GET ABSOLUTE VALUE
C452  12473 017217R       JSB ARINV           IF NUMBER IS NEGATIVE
```

```
C453  12474 073510R        STA T16       SAVE NUMBER
C454  12475 077311R        STB T19
C455  12476 016003X        FSB T12         SUBTRACT EXISTING MAX. VALUE
      12477 013504R
C456  12500 002020         SSA           SKIP AND SWAP IF POSITIVE
C457  12501 026506R        JMP LIN10
C458  12502 063310R        LDA T18       SWAP
C459  12503 067311R        LDB T19
C460  12504 073504R        STA T12
C461  12505 077305R        STB T13
C462  12506 037272R LIN10  ISZ T2        ALL ELEMENTS EXHAUSTED?
C463  12507 026465R        JMP LIN11     NO
C464  12510 063304R        LDA T12       COMPUTE RELATIVE TOLERANCE
C465  12511 067305R        LDB T13       TOL=ABSOLUTE TOL * MAX VALUE
C466  12512 016001X        FMP T16         ABSOLUTE TOLERANCE
      12513 013306R
C467  12514 072057R        STA MLBX1     RELATIVE
C468  12515 076060R        STB MLBX1+1     TOLERANCE
C469  12516 002400         CLA           INITIALIZE PIVOT
C470  12517 073312R        STA LPIV
C471  12520 037274R        ISZ T4        REQUIRE CONSTANT (ROW+1)
C472  12521 037312R LINV1  ISZ LPIV      SELECT NEXT PIVOT
C473  12522 063312R        LDA LPIV      TEST IF HAVE PROCESSED
C474  12523 053274R        CPA T4        LAST PIVOT
C475  12524 126424R        JMP LINV,I    NORMAL EXIT TO MAIN PROG
C476*                                    SCAN PIVOTAL COLUMN FOR
C477*                                    LARGEST ELEMENT
C478  12525 063312R        LDA LPIV      COMPUTE ADDRESS OF PIVOT
C479  12526 067312R        LDB LPIV      COLUMN USING ROUTINE LWHR
C480  12527 073272R        STA T2        ROW COUNTER
C481  12530 017052R        JSB LWHR      ON RETURN, ADDRESS IN A
C482  12531 073271R        STA T1
C483  12532 002400         CLA
C484  12533 073504R        STA T12       T12,T13 IS STORE
C485  12534 073305R        STA T13       FOR GREATEST VALUE
C486  12535 017522R LINV2  JSB DDLD      LOAD FP NUMBER
C487  12536 013271R        DEF T1
C488  12537 002020         SSA           OBTAIN ABSOLUTE VALUE
C489  12540 017217R        JSB ARINV       IF NUMBER IS NEGATIVE
C490  12541 073510R        STA T18       STORE VALUE OF FP NUMBER
C491  12542 077311R        STB T19
C492  12543 016003X        FSB T12         SUBTRACT EXISTING LARGEST VALUE
      12544 013504R
C493  12545 002020         SSA           SKIP AND SWAP IF POSITIVE
C494  12546 026555R        JMP LINV7     T2 STILL CONTAINS MAX VALUE
C495  12547 063310R        LDA T18       STORE NEW MAX VALUE
C496  12550 067311R        LDB T19
C497  12551 073304R        STA T12
C498  12552 077305R        STB T13
C499  12553 063272R        LDA T2        SET T5 TO POSITION IN
C500  12554 073275R        STA T5        COLUMN OF MAX VALUE
C501  12555 037272R LINV7  ISZ T2
C502  12556 063272R        LDA T2        TEST FOR LAST TERM IN COL
C503  12557 053274R        CPA T4
C504  12560 026566R        JMP LINV8     SWAP ROWS
```

```
C505  12561 063273R        LDA T3          COMPUTE
C506  12562 001000          ALS             NEXT ADDRESS
C507  12563 043271R        ADA T1               IN PIVOT
C508  12564 073271R        STA T1                COLUMN
C509  12565 026535R        JMP LINV2       SELECT NEXT TERM
C510*                                       SWAP ROWS LPIV AND T5
C511  12566 063312R LINV8  LDA LPIV        COMPUTE ADDRESS
C512  12567 006404          CLB,INB         OF PIVOTAL ROW
C513  12570 017052R        JSB LWHR
C514  12571 073271R        STA T1          ADDRESS OF PIVOTAL ROW
C515  12572 063275R        LDA T5
C516  12573 006404          CLB,INB
C517  12574 017052R        JSB LWHR
C518  12575 073272R        STA T2          ADDR OF ROW TO BE SWAPPED
C519  12576 063312R        LDA LPIV
C520  12577 006404          CLB,INB         COMPUTE ADDRESS OF
C521  12600 017064R        JSB LWHR2       PIVOTAL ROW IN I-MATRIX
C522  12601 073301R        STA T9
C523  12602 073302R        STA T10         KEEP COPY
C524  12603 063275R        LDA T5
C525  12604 006404          CLB,INB         COMPUTE ADDR OF ROW TO
C526  12605 017064R        JSB LWHR2       BE SWAPPED IN I-MATRIX
C527  12606 073303R        STA T11
C528  12607 063273R        LDA T3
C529  12610 003004          CMA,INA
C530  12611 073304R        STA T12         COUNTER FOR TERMS IN A ROW
C531  12612 017522R LINV3  JSB DDLD      SWAP ONE ELEMENT OF ROW
C532  12613 013271R        DEF T1
C533  12614 073310R        STA T18
C534  12615 077311R        STB T19
C535  12616 017522R        JSB DDLD
C536  12617 013272R        DEF T2
C537  12620 017542R        JSB DDST
C538  12621 013271R        DEF T1
C539  12622 037271R        ISZ T1
C540  12623 037271R        ISZ T1
C541  12624 063310R        LDA T18
C542  12625 067311R        LDB T19
C543  12626 017542R        JSB DDST
C544  12627 013272R        DEF T2
C545  12630 037272R        ISZ T2
C546  12631 037272R        ISZ T2
C547  12632 017522R        JSB DDLD    SWAP 1 ELEMENT OF ROW OF I MAT
C548  12633 013301R        DEF T9
C549  12634 073310R        STA T18
C550  12635 077311R        STB T19
C551  12636 017522R        JSB DDLD
C552  12637 013305R        DEF T11
C553  12640 017542R        JSB DDST
C554  12641 013301R        DEF T9
C555  12642 037301R        ISZ T9
C556  12643 037301R        ISZ T9
C557  12644 063310R        LDA T18
C558  12645 067311R        LDB T19
C559  12646 017542R        JSB DDST
```

```
C560   12647 013303R         DEF T11
C561   12650 037303R         ISZ T11
C562   12651 037303R         ISZ T11
C563   12652 037304R         ISZ T12          SKIP IF DONE
C564   12653 026612R         JMP LINV3        SWAP NEXT ELEMENT
C565*                                         HAVE LARGEST ELEMENT IN
C566*                                         PIVOTAL POSITION. FIND
C567*                                         VALUE AND TEST TO ZERO
C568*                                         FOR SINGULAR MATRIX
C569   12654 063312R         LDA LPIV         COMPUTE
C570   12655 067312R         LDB LPIV           ADDRESS OF
C571   12656 017052R         JSB LWHR             PIVOT
C572   12657 073271R         STA T1                ELEMENT
C573   12660 017322R         JSB DDLD         PIVOT VALUE
C574   12661 013271R         DEF T1
C575   12662 002020          SSA              OBTAIN ABSOLUTE VALUE
C576   12663 017217R         JSB ARINV         IF NUMBER IS NEGATIVE
C577   12664 016003X         FSB MLBX1           SUBTRACT TOLEERANCE AND
       12665 000057R
C578   12666 002020          SSA              COMPARE TO ZERO
C579   12667 016274R         JSB ERROR        PRINT'NEARLY SING MATRIX'
C580*                                         DIVIDE PIVOT ROW AND ROW
C581*                                         IN I-MAT BY PIVOT VALUE
C582   12670 063271R LDUM1   LDA T1           ADDRESS OF PIOT ELEMENT
C583   12671 073272R         STA T2
C584   12672 062661R         LDA HONE         LOAD
C585   12673 064055          LDB .2             1.0
C586   12674 017430R         JSB .FDV.
C587   12675 113271R         DEF T1,I
C588   12676 073310R         STA T10          INVERSE OF PIVOT
C589   12677 077311R         STB T19
C590*                                         MULT ROW BY 1/PIVOT
C591*                                         STARTING AT PIVOT+1
C592   12700 063312R         LDA LPIV
C593   12701 073303R         STA T11          COUNTER FOR ROW
C594   12702 037303R LINV6   ISZ T11          INCREMENT COUNTER
C595   12703 063303R         LDA T11
C596   12704 053274R         CPA T4           TEST FOR END OF ROW
C597   12705 026717R         JMP LIN12
C598   12706 037272R         ISZ T2           ADDRESS OF NEXT ELEMENT
C599   12707 037272R         ISZ T2
C600   12710 017322R         JSB DDLD
C601   12711 013272R         DEF T2
C602   12712 016001X         FMP T16
       12713 013310R
C603   12714 017342R         JSB DDST
C604   12715 013272R         DEF T2
C605   12716 026702R         JMP LINV6
C606*                                         MULT ROW IN I-MATRIX BY
C607*                                         1/PIVOT. SKIP IF ELEMENT=0
C608   12717 063302R LIN12   LDA T10          ADDRESS OF PIVOT ROW
C609   12720 073275R         STA T5           IN I-MATRIX
C610   12721 063273R         LDA T3
C611   12722 003004          CMA,INA  SET
C612   12723 073303R         STA T11          ROW COUNTER
```

```
C613  12724 017322R LIN13 JSB DDLD
C614  12725 013275R       DEF T5
C615  12726 002003        SZA,RSS        SKIP MULTIPLICATION IF ZERO
C616  12727 006002        SZB
C617  12730 026732R       JMP *+2        NOT ZERO
C618  12731 026736R       JMP LIN14      ZERO
C619  12732 016001X       FMP T18
      12733 013310R
C620  12734 017342R       JSB DDST
C621  12735 013275R       DEF T5
C622  12736 037275R LIN14 ISZ T5         NEXT ELEMENT IN I-MATRIX
C623  12737 037275R       ISZ T5
C624  12740 037303R       ISZ T11  DONE?
C625  12741 026724R       JMP LIN13      NO
C626*                                    PERFORM ROW MANIPULATIONS
C627*                                    AND SUBTRACTIONS TO REDUCE
C628*                                    PIVOT COLUMN TO ZERO
C629  12742 002400        CLA
C630  12743 072061R       STA B1
C631  12744 036061R LINV4 ISZ B1    SELECT NEXT ROW
C632  12745 062061R       LDA B1
C633  12746 053274R       CPA T4         TEST FOR LAST ROW
C634  12747 026521R       JMP LINV1      SELECT NEXT PIVOT
C635  12750 053312R       CPA LPIV       TEST TO SKIP PIVOTAL ROW
C636  12751 026744R       JMP LINV4      SKIP PIVOTAL ROW
C637  12752 062061R       LDA B1
C638  12753 006404        CLB,INB
C639  12754 017064R       JSB LWHR2      ADDRESS OF ROW TO BE TRANSFORMED
C640  12755 073303R       STA T11            IN I-MATRIX
C641*                                    COMPUTE MULTIPLIER WHICH
C642*                                    IS THAT ELEMENT IN ROW TO
C643*                                    BE TRANSFORMED WHICH LIES
C644*                                    IN THE PIVOTAL COLUMN
C645  12756 062061R       LDA B1
C646  12757 067312R       LDB LPIV
C647  12760 017052R       JSB LWHR
C648  12761 073301R       STA T9         SAVE ADDRESS
C649  12762 017505R       JSB FTHOP
C650  12763 073277R       STA T7         VALUE OF MULTIPLIER
C651  12764 077300R       STB T8
C652*                                    DO ELIMINATION OF ROWS IN
C653*                                    ORIGINAL MATRIX. START AT
C654*                                    COLUMN LPIV+1
C655  12765 063312R       LDA LPIV
C656  12766 073305R       STA T13        COUNTER
C657  12767 063271R       LDA T1
C658  12770 073272R       STA T2
C659  12771 037305R LINV5 ISZ T13
C660  12772 063305R       LDA T13
C661  12773 053274R       CPA T4         TEST FOR LAST TERM IN ROW
C662  12774 027016R       JMP LIN15
C663  12775 037301R       ISZ T9         T9 IS ADDRESS OF
C664  12776 037301R       ISZ T9         ELEMENT TO BE CHANGED
C665  12777 037272R       ISZ T2         T2 IS ADDR OF CORRESPONDING
C666  13000 037272R       ISZ T2         ELEMENT IN PIVOTAL ROW
```

```
C667   13001 063277R         LDA T7
C668   13002 067300R         LDB T8
C669   13003 017421R         JSB .FMP.
C670   13004 113272R         DEF T2,I
C671   13005 073510R         STA T18          MULTIPLIER*VALUE IN
C001   13006 077311R         STB T19              PIVOT ROW
C002   13007 017322R         JSB DDLD
C003   13010 013301R         DEF T9
C004   13011 016003X         FSB T18
       13012 013310R
C005   13013 017342R         JSB DDST          TRANSFORM ELEMENT
C006   13014 013301R         DEF T9
C007   13015 026771R         JMP LINV5        SELECT NEXT TERM
C008*                                         DO ELIMINATION OF ROWS IN
C009*                                         IDENTITY MATRIX. START AT
C010*                                         BEGINNING OF ROW AND LEAVE
C011*                                         ELEMENT UNCHANGED WHEN ZERO
C012*                                         IN PIVOTAL ROW.
C013   13016 063302R LIN15 LDA T10           ADDRESS OF
C014   13017 073275R         STA T5              PIVOTAL ROW
C015   13020 063273R         LDA T3
C016   13021 003004         CMA,INA          SET
C017   13022 073305R         STA T13            COUNTER
C018   13023 017322R LIN18 JSB DDLD
C019   13024 013275R         DEF T5
C020   13025 037275R         ISZ T5
C021   13026 037275R         ISZ T5
C022   13027 002003         SZA,RSS          SKIP IF ZERO
C023   13030 006002         SZB
C024   13031 027033R         JMP *+2          NOT ZERO
C025   13032 027045R         JMP LIN17        ZERO
C026   13033 016001X         FMP T7       MULTIPLY BY THE MULTIPLIER
       13034 013277R
C027   13035 073510R         STA T10
C028   13036 077311R         STB T19
C029   13037 017322R         JSB DDLD
C030   13040 013303R         DEF T11
C031   13041 016003X         FSB T18
       13042 013310R
C032   13043 017342R         JSB DDST
C033   13044 013303R         DEF T11
C034   13045 037303R LIN17 ISZ T11
C035   13046 037303R         ISZ T11
C036   13047 037305R         ISZ T13
C037   13050 027023R         JMP LIN18        SELECT NEXT TERM
C038   13051 026744R         JMP LINV4        ELIMINATE NEXT ROW
C039*
C040*
C041*********************************************
C042*****   SUBROUTINE LWHR                 *****
C043*********************************************
C044*SUBROUTINE COMPUTES ADDRESS OF AN ELEMENT  *
C045*IN MATRIX GIVEN BY B2. ROW AND COL VALUES  *
C046*ARE SUPPLIED IN A,B. ADDRESS IS LEFT IN A  *
C047*ENTRY LWHR2 COMPUTES ADDR IN MAT B3        *
```

```
C048*************************************************
C049*
C050  13052 000000  LWHR  NOP
C051  13053 077277R       STB T7          SAVE COLUMN #
C052  13054 040052        ADA M1
C053  13055 100200        MPY T3      (A-1)*T3
      13056 013273R
C054  13057 043277R       ADA T7
C055  13060 040052        ADA M1          +(B-1)
C056  13061 001000        ALS
C057  13062 042065R       ADA B2          DDR=B2+2((A-1)*T3+(B-1))
C058  13063 127052R       JMP LWHR,I
C059  13064 000000  LWHR2 NOP
C060  13065 077277R       STB T7
C061  13066 040052        ADA M1
C062  13067 100200         MPY T3
      13070 013273R
C063  13071 043277R       ADA T7
C064  13072 040052        ADA M1
C065  13073 001000        ALS
C066  13074 042065R       ADA B3
C067  13075 127064R       JMP LWHR2,I
C068*
C069**
C070**** RETRIEVE AND STORE  SUB-PROGRAM PARAMETERS0****
C071**
C072*
C073  13076 000000  $ENTR NOP
C074  13077 073121R       STA SORCE
C075  13100 167076R       LDB $ENTR,I
C076  13101 077122R       STB DEST
C077  13102 007004        CMB,INB
C078  13103 047076R       ADB $ENTR
C079  13104 044051        ADB M2
C080  13105 007004        CMB,INB
C081  13106 077123R       STB CNTR
C082  13107 037076R       ISZ $ENTR
C083  13110 163121R .LOOP LDA SORCE,I
C084  13111 173122R       STA DEST,I
C085  13112 063121R       LDA SORCE
C086  13113 040052        ADA M1
C087  13114 073121R       STA SORCE
C088  13115 037122R       ISZ DEST
C089  13116 037123R       ISZ CNTR
C090  13117 027110R       JMP .LOOP
C091  13120 127076R       JMP $ENTR,I
C092  13121 000000  SORCE NOP
C093  13122 000000  DEST  NOP
C094  13123 000000  CNTR  NOP
C095  13124 000000  EXECR NOP         GET RECORD FROM CRT/TICKY-TICKY
C096  13125 077260R       STB BUFFR
C097  13126 073267R       STA BUFFL
C098  13127 063124R       LDA EXECR
C099  13130 073270R       STA RET
C100  13131 060054        LDA .1
```

```
C101   13132 073265R        STA RCODE
C102   13133 063136R        LDA D401
C103   13134 073266R        STA CONWD
C104   13135 027254R        JMP IEXEC
C105   13136 000401   D401  OCT 401
C106   13137 000000   EXECI NOP              GET RECORD FROM PHOTO READER
C107   13140 077260R        STB BUFFR
C108   13141 003004         CMA,INA
C109   13142 073267R        STA BUFFL
C110   13143 063137R        LDA EXECT
C111   13144 073270R        STA RET
C112   13145 060054         LDA .1
C113   13146 073265R        STA RCODE
C114   13147 060060         LDA .5
C115   13150 073266R        STA CONWD
C116   13151 027254R        JMP IEXEC
C117   13152 000000   EXECS NOP       SET END OF TAPE
C118   13153 016005X        JSB EXEC
C119   13154 013157R        DEF *+3
C120   13155 000056         DEF .3
C121   13156 013160R        DEF CON2
C122   13157 127152R        JMP EXECS,I
C123   13160 000705   CON2  OCT 000705
C124*
C125***  ENTER WITH CHARACTER COUNT IN THE 'A' REGISTER
C126***  AND THE BUFFER ADDRESS IN THE 'B' REGISTER.
C127*
C128   13161 000000   EXECW NOP              WRITE RECORD ON CRT
C129   13162 077260R        STB BUFFR
C130   13163 002021         SSA,RSS
C131   13164 003004         CMA,INA
C132   13165 073267R        STA BUFFL
C133   13166 063161R        LDA EXECW
C134   13167 073270R        STA RET
C135   13170 060055         LDA .2
C136   13171 073265R        STA RCODE
C137   13172 102501         LIA 01
C138   13173 012261R        AND HALF
C139   13174 067264R        LDB XIT
C140   13175 006021         SSB,RSS
C141   13176 027201R        JMP *+3
C142   13177 002002         SZA
C143   13200 027575R        JMP RDYPT
C144   13201 102501         LIA 01
C145   13202 012263R        AND MNEG
C146   13203 002002         SZA
C147   13204 060061         LDA .6
C148   13205 002003         SZA,RSS
C149   13206 060054         LDA .1
C150   13207 073266R        STA CONWD
C151   13210 027254R        JMP IEXEC
C152*
C153***(ADDED 7-APR-71)
C154*
C155   13211 000000   LPUN  NOP
```

```
C156   13212 006002          SZB
C157   13213 027232R         JMP LPUN1
C158   13214 062252R         LDA D133
C159   13215 073231R         STA CNTHO
C160   13216 016005X LPUN3   JSB EXEC      PUNCH RECORD ON H.S. PUNCH
C161   13217 013224R         DEF *+5
C162   13220 000055          DEF .2
C163   13221 013227R         DEF CON1
C164   13222 013230R         DEF BUFR1
C165   13223 000054          DEF .1
C166   13224 037231R         ISZ CNTHO
C167   13225 027216R         JMP LPUN3
C168   13226 127211R         JMP LPUN,I
C169   13227 000104  CON1    OCT 104
C170   13230 000000  BUFR1   NOP
C171   13231 000000  CNTHO   NOP
C172   13232 077260R LPUN1   STB BUFFR
C173   13233 003004          CMA,INA
C174   13234 073267R         STA BUFFL
C175   13235 063211R         LDA LPUN
C176   13236 073270R         STA RET
C177   13237 060057          LDA .4
C178   13240 073266R         STA CONWD
C179   13241 027252R         JMP LPUN2
C180   13242 000000  EXECP   NOP           LIST RECORD ON LINE PRINTER
C181   13243 077260R         STB BUFFR
C182   13244 003004          CMA,INA
C183   13245 073267R         STA BUFFL
C184   13246 063242R         LDA EXECP
C185   13247 073270R         STA RET
C186   13250 060061          LDA .6
C187   13251 073266R         STA CONWD
C188   13252 060055  LPUN2   LDA .2
C189   13253 073265R         STA RCODE
C190                         EXT EXEC
C191   13254 016005X IEXEC   JSB EXEC
C192   13255 013262R         DEF *+5
C193   13256 013265R         DEF RCODE
C194   13257 013266R         DEF CONWD
C195   13260 000000  BUFFR   NOP
C196   13261 013267R         DEF BUFFL
C197   13262 060001          LDA 1
C198   13263 127270R         JMP RET,I
C199*
C200*
C201   13264 000000  XIT     BSS 1
C202   13265 000000  RCODE   BSS 1
C203   13266 000000  CONWD   BSS 1
C204   13267 000000  BUFFL   BSS 1
C205   13270 000000  RET     BSS 1
C206*
C207*
C208*****************************************
C209*              CONSTANTS                *
C210*****************************************
```

```
C211*
C212   13271 000000  T1     BSS 1          TEMPORARY CONSTANTS
C213   13272 000000  T2     BSS 1
C214   13273 000000  T3     BSS 1
C215   13274 000000  T4     BSS 1
C216   13275 000000  T5     BSS 1
C217   13276 000000  T6     BSS 1
C218   13277 000000  T7     BSS 1
C219   13300 000000  T8     BSS 1
C220   13301 000000  T9     BSS 1
C221   13302 000000  T10    BSS 1
C222   13303 000000  T11    BSS 1
C223   13304 000000  T12    BSS 1
C224   13305 000000  T13    BSS 1
C225   13306 041433  T16    DEC +1E-6      ABSOLUTE TOLERANCE
       13307 157733
C226   13310 000000  T18    BSS 1
C227   13311 000000  T19    BSS 1
C228   13312 000000  LPIV   BSS 1
C229   13313 000000  FINIS  BSS 700
C230   14607 000000  LWAMA  BSS 1
C231   14610 000000  SBUFB  BSS 97
C232   14751 000000  SYMTB  BSS 64
C233**
C234***  PERFORM LDA 1,1
C235**
C236   15051 000000  LDB1   NOP
C237   15052 060001         LDA B
C238   15053 017204R        JSB DREAD
C239   15054 127051R        JMP LDB1,I
C240**
C241***   PERFORM STA 1,I
C242**
C243   15055 000000  STB1   NOP
C244   15056 077130R        STB TMP2
C245   15057 017151R        JSB DSTOR
C246   15060 015130R        DEF TMP2
C247   15061 127055R        JMP STB1,I
C248**
C249***   PERFORM LDA TEMPS,I
C250**
C251   15062 000000  LDTMP  NOP
C252   15063 062045R        LDA TEMPS
C253   15064 017204R        JSB DREAD
C254   15065 127062R        JMP LDTMP,I
```

```
C256*
C257*****************************************************************
C258****          DISK BUFFER SUBROUTINES                        ******
C259*****************************************************************
C260*  ADDRESS CONVERSION SUBROUTINE
C261*     ENTER WITH ADDRESS IN A REG
C262*
C263  15066 000000  ACON NOP
C264  15067 006400        CLB
C265  15070 077121R       STB SCNT    SET SECTOR COUNT TO ZERO
C266  15071 066011C       LDB FTRAK    common (see p 2)
C267  15072 077122R       STB TCNT     SET TRACK COUNT TO START OF W.A.
C268  15073 002020        SSA            ADDR > 32767
C269  15074 016274R       JSB ERROR p7       NO, ERROR!
C270  15075 043126R DIA1  ADA M3072            COMPUTE TRACK ADDR.
C271  15076 002021        SSA,RSS
C272  15077 027105R       JMP INCTR
C273  15100 043124R       ADA .3072
C274  15101 043125R DIA2  ADA M128             COMPUTE SECTOR ADDR.
C275  15102 002021        SSA,RSS
C276  15103 027107R       JMP INCSC
C277  15104 127066R       JMP ACON,I
C278*
C279*
C280  15105 037122R INCTR ISZ TCNT
C281  15106 027075R       JMP DIA1
C282*
C283  15107 037121R INCSC ISZ SCNT
C284  15110 067121R       LDB SCNT
C285  15111 057135R       CPB SIZE
C286  15112 017114R       JSB ITRK
C287  15113 027101R       JMP DIA2
C288*
C289  15114 000000  ITRK NOP
C290  15115 037122R       ISZ TCNT
C291  15116 006400        CLB
C292  15117 077121R       STB SCNT
C293  15120 127114R       JMP ITRK,I
C294*
C295*
C296  15121 000000  SCNT  BSS 1      sector count
C297  15122 000000  TCNT  BSS 1      track count
C298  15123 000000  TMP   BSS 1
C299  00172         .128  EQU 8200
C300  15124 006000  .3072 DEC 3072
C301  15125 177600  M128  DEC -128
C302  15126 172000  M3072 DEC -3072
C303  15127 000000  TMP1 BSS 1
C304  15130 000000  TMP2 BSS 2
C305  15132 000000  TMP3 BSS 1
C306  15133 000000  TMP4 BSS 1
C307  15134 000000  TMP5 BSS 1
C308  15135 000000  SIZE  BSS 1          # OF SECTORS PER DISK TRACK
C309  15136 000102  CW1   OCT 102
C310*
```

```
C311*    SUB TO READ INTERMEDIATE CODE FROM DISK BUFFER
C312*
C313  15137 000000  READ1 NOP
C314  15140 016005X        JSB EXEC
C315  15141 015150R        DEF *+7
C316  15142 000054         DEF .1
C317  15143 015136R        DEF CW1 p156
C318  15144 100004C        DEF BGNPG,I  common, p2

FG 024
LN  0319        DEF BFRL
C319  15145 000000         DEF BFRL
C320  15146 015122R        DEF TCNT } p156
C321  15147 015121R        DEF SCNT }
C322  15150 127137R        JMP READ1,I
C323**
C324*    STORE INTERMEDIATE CODE AT VIRTUAL ADDRESS
C325**
C326  15151 000000  DSTOR NOP
C327  15152 077123R        STB TMP          SAVE A AND B
C328  15153 073127R        STA TMP1         PROGRAM PARAMETERS
C329  15154 163151R        LDA DSTOR,I
C330  15155 160000         LDA 0,1          GET VIRTUAL ADDR
C331  15156 037151R        ISZ DSTOR        INCREMENT RETURN (P+2)
C332  15157 073133R        STA TMP4         SAVE VIRTUAL ADDRESS
C333  15160 017213R        JSB FNDPG        FIND DISK PAGE CORRES. TO ADDR.
C334  15161 066001C        LDB PGPT1        BOTH CORE PAGES SAME?
C335  15162 056002C        CPB PGPT2
C336  15163 027171R        JMP DBLWR        YES, WRITE DATA TO BOTH
C337  15164 067127R        LDB TMP1         NO, WRITE TO 1
C338  15165 174000         STB 0,1
C339  15166 067123R        LDB TMP          RESTORE (A) & (B)
C340  15167 063127R        LDA TMP1
C341  15170 127151R        JMP DSTOR,I
C342*
C343  15171 062001C DBLWR  LDA PGPT1       WRITE SAME INFO TO BOTH
C344  15172 003004         CMA,INA          PAGES
C345  15173 043133R        ADA TMP4
C346  15174 042005C        ADA PG1
C347  15175 067127R        LDB TMP1         GET THE DATA
C348  15176 174000         STB 0,1          STORE IN FIRST PAGE

FG 157
LN  0349        ADA BFRL        INCREMENT ADDR TO PAGE 2
C349  15177 042000         ADA BFRL         INCREMENT ADDR TO PAGE 2
C350  15200 174000         STB 0,I          STORE IN SECOND PAGE
C351  15201 067123R        LDB TMP          RESTORE (A) & (B)
C352  15202 063127R        LDA TMP1
C353  15203 127151R        JMP DSTOR,I
C354***
C355**      READ INTERMEDIATE CODEVIRTUAL ADDRESS
C356***
C357  15204 000000  DREAD NOP
C358  15205 077123R        STB TMP          SAVE B
C359  15206 073133R        STA TMP4         SAVE VIRTUAL ADDR
```

```
C360  15207 017215R        JSB FNDPG       FIND DISK PAGE CORRES. TO ADDR.
C361  15210 160000         LDA 0,I         PICK UP DATA AT VIRTUAL ADDRESS
C362  15211 067123R        LDB TMP         RESTORE B
C363  15212 127204R        JMP DREAD,I
C364***
C365**   FIND AND LOAD DISK PAGE CORRESPONDING TO
C366**   VIRTUAL ADDRESS
C367***
C368  15213 000000  FNDPG NOP
C369  15214 063133R        LDA TMP4      V.A. < END OF IM CODE STORAGE?
C370  15215 003000         CMA
C371  15216 042010C        ADA PBPTR
C372  15217 002020         SSA
C373  15220 027251R        JMP FND2   NO, SET POINTERS FOR PAGE 2
C374  15221 027242R        JMP FND1   YES, SET POINTERS FOR PAGE 1
C375  15222 063133R FND3   LDA TMP4
C376  15223 003000         CMA             VIRTUAL ADDRESS >=
C377  15224 042000C        ADA PGPTR         BEGINNING OF CURRENT PAGE
C378  15225 002021         SSA,RSS
C379  15226 017260R        JSB SWAP        NO, GET PROPER PAGE
C380  15227 063133R        LDA TMP4        YES, VIRTUAL ADDRESS >
C381  15230 003000         CMA                END OF CURRENT PAGE?
C382  15231 042000C        ADA PGPTR

PG 157
LN  0383        ADA BFRL
C383  15232 042000         ADA BFRL
C384  15233 002020         SSA
C385  15234 017260R        JSB SWAP         YES, GET PROPER PAGE
C386  15235 062000C        LDA PGPTR              NO, COMPUTE
C387  15236 003004         CMA,INA
C388  15237 043133R        ADA TMP4         CORE (DISK) BUFFER ADDRESS
C389  15240 042004C        ADA BGNPG
C390  15241 127213R        JMP FNDPG,I
C391  15242 062001C FND1   LDA PGPT1
C392  15243 072000C        STA PGPTR
C393  15244 062005C        LDA PG1
C394  15245 072004C        STA BGNPG
C395  15246 062004R        LDA PT1
C396  15247 072003C        STA PGPT
C397  15250 027222R        JMP FND3
C398  15251 062002C FND2   LDA PGPT2
C399  15252 072000C        STA PGPTR
C400  15253 062006C        LDA PG2
C401  15254 072004C        STA BGNPG
C402  15255 062005R        LDA PT2
C403  15256 072003C        STA PGPT
C404  15257 027222R        JMP FND3
C405***
C406**    FIND AND LOAD DISK PAGES
C407***
C408  15260 000000  SWAP   NOP
C409  15261 002400         CLA           INITIALIZE PAGE POINTER
C410  15262 073134R SWP1   STA TMP5
C411  15263 003004         CMA,INA
```

```
C412  15264 043133R        ADA TMP4    VIRTUAL ADDR. > PAGE POINTER?
C413  15265 002020         SSA
C414  15266 027272R        JMP SWP2      NO, EXIT WITH ADDR
C415  15267 063134R        LDA TMP5     YES, INCREMENT PAGE POINTER

FG 158
LN  0415        ADA BFRL     BY BUFFER LENGTH
C416  15270 042000         ADA BFRL     BY BUFFER LENGTH
C417  15271 027262R        JMP SWP1     CHECK AGAIN

FG 159
LN  0418  SWP2  LDA BFRL     DECREMENT PAGE POINTER BY
C418  15272 062000  SWP2    LDA BFRL     DECREMENT PAGE POINTER BY
C419  15273 003004         CMA,INA      1 BUFFER LENGTH
C420  15274 043134R        ADA TMP5
C421  15275 073134R        STA TMP5     AND SAVE
C422  15276 062000C        LDA PGPTR    GET ADDRESS OF CURRENT CORE RES PAGE
C423  15277 017066R        JSB ACUN
C424  15300 016005X        JSB EXEC     RETURN TO DISK
C425  15301 015310R        DEF *+7
C426  15302 000055         DEF .2
C427  15303 015136R        DEF CW1
C428  15304 100004C        DEF BGNPG,I

FG 159
LN  0429        DEF BFRL
C429  15305 000000         DEF BFRL
C430  15306 015122R        DEF TCNT
C431  15307 015121R        DEF SCNT
C432  15310 063134R        LDA TMP5     GET ADDRESS OF NEW CORE RES PAGE
C433  15311 172003C        STA PGPT,I    STORE IN POINTER
C434  15312 072000C        STA PGPTR
C435  15313 017066R        JSB ACUN
C436  15314 017137R        JSB READ1     FETCH PAGE FROM DISK
C437  15315 127260R        JMP SWAP,I
C438  15316 000000  SVOP1  BSS 2
C439  15320 000000  SVOP2  BSS 2
C440*****************************************
C441**** SUBROUTINE DISK DOUBLE LOAD*******
C442*****************************************
C443*   PERFORM DLD ADR,I
C444*
C445*  CALLING SEQUENCE:  JSB DDLD
C446*                     DEF ADR
C447*
C448*****************************************
C449*
C450  15322 000000  DDLD NOP
C451  15323 163522R        LDA DDLD,I      GET DISK ADR
C452  15324 160000         LDA A,I
C453  15325 012222R        AND INF     REMOVE V.A. BIT
C454  15326 073340R        STA VA           AND SAVE
C455  15327 037522R        ISZ DDLD         INCR RETURN (P+2)
C456  15330 017204R        JSB DREAD
C457  15331 073341R        STA SV.LO
```

```
C458  15332 037340R        ISZ VA              INCR DISK ADR
C459  15333 063340R        LDA VA
C460  15334 017204R        JSB DREAD
C461  15335 064000          LDB A
C462  15336 063341R        LDA SV.LO
C463  15337 127322R        JMP DDLD,I      RETURN
C464*
C465  15340 000000  VA   BSS 1
C466  15341 000000  SV.LO  BSS 1



C468**************************************************
C469*****SUBROUTINE DISK DOUBLE STORE**************
C470**************************************************
C471*
C472*      PERFORM   DST ADR,I
C473*
C474*      CALLING SEQUENCE:      JSB DDST
C475*                              DEF ADR
C476*
C477**************************************************
C478*
C479  15342 000000  DDST  NOP
C480  15343 104400        DST SVOP1         SAVE OPERAND
      15344 015516R
C481  15345 163342R        LDA DDST,I           GET DISK ADR
C482  15346 160000        LDA A,I
C483  15347 012222R        AND INF
C484  15350 073340R        STA VA             AND SAVE
C485  15351 037342R        ISZ DDST          INCR RETURN (P+2)
C486  15352 063316R        LDA SVOP1
C487  15353 017151R        JSB DSTOR
C488  15354 015340R        DEF VA
C489  15355 037340R        ISZ VA
C490  15356 063317R        LDA SVOP1+1
C491  15357 017151R        JSB DSTOR
C492  15360 015340R        DEF VA
C493  15361 104200        DLD SVOP1         RESTORE (A) & (B)
      15362 015516R
C494  15363 127342R        JMP DDST,I
C495**************************************
C496***     SUBROUTINE FETCH PARAMETER   ***
C497**************************************
C498*
C499*    FETCHES PARAMETERS AT ADDRESSES
C500*    EITHER ON DISK OR IN CORE
C501*
C502*    CALLING SEQUENCE:    JSB F/PAR
C503*                         DEF ADR
C504*
C505*    WHERE ADR IS THE ADDRESS OF THE PARAMETER POINTER
C506*
C507**************************************
```

```
C508*
C509  15364 000000  F/PAR NOP
C510  15365 163364R       LDA F/PAR,I
C511  15366 037364R       ISZ F/PAR
C512  15367 160000        LDA A,I
C513  15370 070001        STA B        GET PARAMETER ADDRESS AND SAVE
C514  15371 012222R       AND INF
C515  15372 072367R       STA P/ADR
C516  15373 006021        SSB,RSS
C517  15374 027400R       JMP *+4
C518  15375 017322R       JSB DDLD   YES,  FETCH PAR FROM DISK
C519  15376 000367R       DEF P/ADR
C520  15377 127364R       JMP F/PAR,I
C521  15400 104200        DLD P/ADR,I    FETCH PAR FROM CORE
      15401 100367R
C522  15402 127364R       JMP F/PAR,I
C523  00367          P/ADR EQU MVTOC
C524*             ************************
C525*             SUBROUTINE FLOATING ADD
C526*             ************************
C527*
C528*   EXECUTE FAD ADR,I
C529*
C530*       ADR= ADDRESS OF 2ND OPERAND. MAY BE
C531*            IN CORE OR ON DISK
C532*       CALLING SEQUENCE:
C533*
C534*                          JSB .FAD.
C535*                          DEF ADR
C536*
C537*       IF BIT #15 OF ADR IS SET 2ND
C538*       OPERAND IS ASSUMED TO BE ON DISK.
C539*
C540*          ************************
C541  15403 000000  .FAD. NOP
C542  15404 104400        DST SVOP1       SAVE OPERAND # 1
      15405 015516R
C543  15406 002400        CLA             SELECT ADD OP.
C544  15407 017437R       JSB ARITH       GO DO IT
C545  15410 037405R       ISZ .FAD.       INCREMENT RETURN
C546  15411 127405R       JMP .FAD.,I
C547*
C548*             ******************************
C549*             SUBROUTINE FLOATING SUBTRACT
C550*             ******************************
C551*
C552*     SEE FLOATING ADD FOR DESCRIPTION
C553*
C554*             ******************************
C555*
C556  15412 000000  .FSB. NOP
C557  15413 104400        DST SVOP1       SAVE OPERAND # 1
      15414 015516R
C558  15415 060056        LDA .3          SELECT SUBTRACT OP.
C559  15416 017437R       JSB ARITH       GO DO IT
```

```
C560  15417 037412R        ISZ .FSB.       INCREMENT RETURN
C561  15420 127412R        JMP .FSB.,I
C562*
C563*            ******************************
C564*            SUBROUTINE FLOATING MULTIPLY
C565*            ******************************
C566*
C567*    SEE FLOATING ADD FOR DESCRIPTION
C568*
C569*            ******************************
C570*
C571  15421 000000  .FMP. NOP
C572  15422 104400        DST SVOP1       SAVE OPERAND # 1
      15423 015516R
C573  15424 060061        LDA .6          SELECT MULTIPLY OP.
C574  15425 017437R        JSB ARITH       GO DO IT
C575  15426 037421R        ISZ .FMP.       INCREMENT RETURN
C576  15427 127421R        JMP .FMP.,I
C577*
C578*            ****************************
C579*            SUBROUTINE FLOATING DIVIDE
C580*            ****************************
C581*
C582*    SEE FLOATING ADD FOR DESCRIPTION
C583*
C584*            ****************************
C585*
C586  15430 000000  .FDV. NOP
C587  15431 104400        DST SVOP1       SAVE OPERAND # 1
      15432 015516R
C588  15433 060064        LDA .9          SELECT DIVIDE OP.
C589  15434 017437R        JSB ARITH       GO DO IT
C590  15435 037430R        ISZ .FDV.       INCREMENT RETURN
C591  15436 127430R        JMP .FDV.,I
C592*
C593*            ******************************
C594*            GENERAL ARITHMETIC SUBROUTINE
C595*            ******************************
C596*
C597*    EXECUTES AN ARITHMETIC OPERATION BETWEEN
C598*    DATA IN THE A & B REGISTERS AND DATA STORED
C599*    IN CORE OR ON THE DISK. THE RESULT IS RETURNED
C600*    IN THE A & B REGISTERS.
C601*
C602*            ******************************
C603*
C604  15437 000000  ARITH NOP
C605  15440 043504R        ADA .TADR       COMPUTE ADDRESS OF ARITH OP.
C606  15441 073467R        STA OPMOD       AND SAVE IT
C607  15442 063437R        LDA ARITH       GET ADDRESS OF
C608  15443 040046        ADA M5                PARAMETER
C609  15444 160000        LDA A,I                 ADDRESS
C610  15445 160000        LDA A,I
C611  15446 002021        SSA,RSS         MAT. FLAG SET?
C612  15447 027453R        JMP ARTH0       NO, PROCESS NORMALLY
```

```
C613  15450 012222R        AND INF          YES, 2ND OPERAND IS ON DISK
C614  15451 160000         LDA A,I          GET PARAMETER ADDRESS
C615  15452 027456R        JMP ARTH3        GO TO GET 2ND OPERAND
C616  15453 160000  ARTH0  LDA A,I          GET PARATER ADDRESS
C617  15454 002021         SSA,RSS          2ND OPERAND ON DISK
C618  15455 027460R        JMP ARTH1        NO, FETCH IT FROM CORE
C619  15456 017505R ARTH3  JSB FTHOP        YES, FETCH IT FROM DISK
C620  15457 027462R        JMP ARTH2
C621  15460 104200  ARTH1  DLD A,I          FETCH 2ND OPERAND FROM CORE
      15461 100000
C622  15462 104400  ARTH2  DST SVOP2        SAVE TEMPORARILY
      15463 015320R
C623  15464 104200         DLD SVOP1        GET 1ST OPERAND
      15465 015516R
C624  15466 127467R        JMP *+1,I        GO PERFORM OPERATION
C625  15467 000000  OPMOD  NOP
C626  15470 016002X        FAD SVOP2
      15471 015320R
C627  15472 127437R        JMP ARITH,I
C628  15473 016003X        FSB SVOP2
      15474 015320R
C629  15475 127437R        JMP ARITH,I
C630  15476 016001X        FMP SVOP2
      15477 015320R
C631  15500 127437R        JMP ARITH,I
C632  15501 016004X        FDV SVOP2
      15502 015320R
C633  15503 127437R        JMP ARITH,I
C634  15504 015470R .TADR  DEF OPMOD+1
C635**
C636***    FETCH OPERAND FROM DISK TO A AND B
C637**
C638**
C639***    ADDRESS OF OPERAND INITIALLY IN A REGISTER
C640**     PERFORMS FOLLOWING OPERATIONS
C641**         STA B
C642**         LDA B,I
C643**         INB
C644**         LDB B,I
C645**
C646**     ALL ADDRESS RELATE TO DISK BUFFER VIRTUAL ADDRESS
C647**
C648  15505 000000  FTHOP NOP
C649  15506 012222R        AND INF     REMOVE D.A. BIT
C650  15507 070001         STA B
C651  15510 017051R        JSB LDBI
C652  15511 073130R        STA TMP2
C653  15512 006004         INB
C654  15513 017051R        JSB LDBI
C655  15514 064000         LDB A
C656  15515 063130R        LDA TMP2
C657  15516 127505R        JMP FTHOP,I
C658**
C659**
C660  15517 000000  LDOP2 NOP
```

```
C661  15520 063152R        LDA BIN02
C662  15521 002020         SSA
C663  15522 027525R        JMP *+3
C664  15523 160000         LDA A,I
C665  15524 127517R      * JMP LDOP2,I
C666  15525 017204R        JSB DREAD
C667  15526 127517R        JMP LDOP2,I
C668                       END ENTRY
```

FG 159
0007 ERRORS *TOTAL **RTE ASMB 92067-16011

/XREF : TABLE OVERFLOW